AD NUMBER

AD472864

NEW LIMITATION CHANGE

TO

Approved for public release, distribution unlimited

FROM

Distribution authorized to U.S. Gov't. agencies and their contractors; Administrative/Operational Use; Sep 1965. Other requests shall be referred to Air Force Weapons Lab., Kirtland AFB, NM 87117.

AUTHORITY

Air Force Weapons Lab ltr dtd 1 Jun 1972

SECURITY MARKING

The classified or limited status of this report applies to each page, unless otherwise marked.
Separate page printouts MUST be marked accordingly.

DYNAMIC STRESSES IN A THICK ELASTIC CYLINDER SUBJECT TO TRANSIENT PRESSURE LOADINGS

Volume II: Discussion of Computer Program

Frank Nolan

Grumman Aircraft Engineering Corporation
Bethpage, N. Y.

Contract AF29(601)-5993

TECHNICAL REPORT NO. AFWL-TR-65-20, Vol. II

September 1965

AIR FORCE WEAPONS LABORATORY
Research and Technology Division
Air Force Systems Command
Kirtland Air Force Base
New Mexico

Research and Technology Division
AIR FORCE WEAPONS LABORATORY
Air Force Systems Command
Kirtland Air Force Base
New Mexico

AFWL-TR-65-20, Vol. II

DYNAMIC STRESSES IN A THICK ELASTIC CYLINDER SUBJECT TO TRANSIENT PRESSURE LOADINGS

Volume II: Discussion of Computer Program

Frank Nolan

Grumman Aircraft Engineering Corporation
Bethpage, N. Y.

Contract AF 29(601)-5993

TECHNICAL REPORT NO. AFWL-TR-65-20, Vol. II.

FOREWORD

This report was prepared by Grumman Aircraft Engineering Corporation, Bethpage, N.Y. under Contract AF 29(601)-5993. The work was performed under Program Element 7.60.06.01 D, Project 5710, Subtask 13.148, and was funded by the Defense Atomic Support Agency (DASA). Inclusive dates of research were 1 June 1963 through 16 March 1965. The report was submitted on 3 September 1965.

The report is presented in two volumes. Volume I is concerned with the theoretical analysis and discussion of the solution. Volume II, subtitled "Computer Program," presents a listing and discusses the details of the computer program.

The authors extend their appreciation to the former Project Officer, Lt Joe E. Johnson (WLDC) for continuous cooperation during the course of the research.

This report has been reviewed and is approved.

DWAYNE D. PIEPENBURG
1Lt USAF
Project Officer

ROBERT E. CRAWFORD
Major USAF
Deputy Chief,
Civil Engineering Branch

JOHN W. KODIS
Colonel USAF
Chief, Development Division

ABSTRACT

The response of a hollow circular cylindrical shell of arbitrary thickness, in either an elastic or a viscoelastic medium, to transient dilatational and shear waves (and their superposition) is presented. The solution is valid within the scope of the linear theory of elasticity or viscoelasticity. The technique for obtaining the solution relies upon 1) the construction of a train of incident pulses from steady state components, where each pulse represents the time history of the transient stress in the incident wave, and 2) the existence of a physical mechanism that, between pulses, restores the disturbed particles of the cylinder and the surrounding medium to an unstrained state of rest.

The influence on the cylinder response of the following factors is discussed: liner thickness, cylinder-medium impedance mismatch, viscoelasticity in the medium, and incident wave form (step pulse, rectangular, triangular, linear rise-exponential decay).

CONTENTS

SECTION I

GENERAL DISCUSSION OF PROGRAM

A. Program Logic

The program presents a rather formidable appearance, principally because of its length, but the central idea is quite simple - for each of the triples (ρ, θ, $\overline{x}$) specified by the input, the terms of a double series are computed and summed. Actually there are five distinct series for each (ρ, θ, $\overline{x}$), and since each series is a double series, the individual terms can be regarded as dependent on six parameters, viz, ρ, θ, and $\overline{x}$, two summation indices, and a final index to distinguish among the five sums. It is this six-way indexing that lies at the root of the difficulties encountered in preparing the program.

Perhaps the most straightforward approach would have run along the following lines: choose ρ, θ, and $\overline{x}$; then generate the successive terms of the sum for ρ, θ, and $\overline{x}$ and perform the addition. Of course, five sums must be provided for, but this could be done without undue complication. The decisive consideration against such a procedure is the unacceptably high amount of computer time that would be required. The calculation of an individual term has much in common with that of many other terms, and it would be pointless to ignore the possibility of exploiting the similarity. Although we have not attempted to form a close estimate of the time-saving made possible by the procedure we adopted, a factor of 100 appears to be conservative. Denoting the summation indices by p and n, the over-all arrangement is as follows: p is fixed, then ρ, and then for each n we form the terms of the five sums associated with each θ and $\overline{x}$ and sum over n; ρ then takes on its next value and when ρ has attained its final value, p is incremented. If k is the index used to distinguish among the five sums, then the order of variation of parameters is p, ρ, n, θ, $\overline{x}$, and k, with the understanding that p is the "most permanent" index and k the most rapidly changing. This arrangement is not necessarily claimed to be optimal, but it seemed to represent a feasible compromise between the (sometimes conflicting) requirements of speed and convenient use of storage.

Our discussion of the program is confined to a few major items in the belief that a detailed description would be inordinately difficult to follow. As a supplement, we have included two flow charts (at different levels of detail) to facilitate understanding of the flow of control.

It is widely recognized that the accurate summation of a convergent series is a severe problem. Care must be taken that enough terms are added and also that the computed partial sums bear some resemblance to the corresponding exact partial sums. In a strict sense, both problems are insoluble in a general setting; for a computer program can "see" only a finite number of terms, and unless these terms are in some sense "typical" one cannot be certain that significant terms may not have been omitted. Furthermore, the computed sum of even three numbers may bear no relation to the true sum, so that even when a particular partial sum is itself adequate, the computed partial sum may have no correct digits. These considerations indicate a need for caution, and we have acted accordingly.

Our program faced two additional complications in that the series involved are double series, and in the need for "simultaneous" summation of a number of series. In describing the process, we attempt to maintain a degree of clarity by momentarily neglecting the fact that many series are being treated at once.

There are five input parameters that govern the summation of the series. These are NACRCY, KAPP, MØTAP,KAPN, and MØTAN. NACRCY stipulates a negative power of 10 that is to be an approximate bound of the relative error (NACRCY = 3 indicates, for example, that an attempt is to be made to keep the relative error below the level of 10^{-3}, or roughly, to attain three correct significant digits). KAPP is the maximum value that p (a summation index) will be allowed (if satisfactory convergence appears to take place earlier, p may stop short of KAPP). MØTAP is the number of consecutive "small" terms required to halt the addition ("small" is defined later). For each p, we sum over the index n, up to a maximum of KAPN terms, with the provision that if MØTAN consecutive "small" terms appear first, the summation over n is stopped.

Associated with each double series is an integer that plays a dual role. For present purposes, designate the series by D2 and the corresponding integer by K2. The primary function of K2 is to preserve a record of the maximum size reached by any of the terms or partial sums of D2. Explicitly, K2 starts at zero, and during the summation it maintains a value of $128(m + 128)$ where 2^m is the smallest power of 2 which is larger than the terms and partial sums of D2 so far encountered. The number 128, which is itself a power of 2, is

convenient for use on a binary machine, and the multiplication of m + 128 by 128 serves to zero out the low-order 7 binary digits of K2: these digit positions are used for another purpose. A newly generated term is regarded as "small" if either of two conditions obtains:

1) the term is smaller than TØL times the corresponding partial sum; here TØL = $.1 \times 10^{-NACRCY}$

2) the term is smaller than 10^{-7} times the power of 2 (m) specified by K2. For "row" summation (i.e., over the index n, rather than p), the factor $.5 \times 10^{-7}$ is used in place of 10^{-7}.

Before elaborating on this definition of "small", we note that the secondary role of K2, for which its low-order 7 binary digits are used, is to keep track of the number of consecutive small terms encountered; when this integer reaches MØTAP (or MØTAN for n-summation), the addition and generation of terms for the sum is stopped.

Roughly speaking, Criterion 1) states that the term has no serious influence on the corresponding partial sum. When this is true of several consecutive terms, it furnishes evidence that the remainder is small, or that the partial sum is adequate; an extra factor of .1 is included so as to be conservative. Criterion 2) is an assertion that the term is negligible compared to some of the numbers to which it is added - not necessarily the most recent partial sum, which may (conceivably) be substantially smaller than some that precede it. The intent of this condition is to avoid the inclusion of many additional terms in a futile pursuit of better accuracy; for relatively low values of NACRCY, Criterion 1) will ordinarily come into play before 2), but if NACRCY is 6 or 7, the second criterion is more realistic and should be dominant.

When all the summation is complete, these "descriptive integers", i.e., the K2's, are redefined so as to estimate the relative accuracy obtained. The sign is positive if the p-series convergence appeared satisfactory, i.e., if MØTAP consecutive small terms were found; otherwise the sign is negative. The magnitude of the new K2 cannot be greater than NACRCY and is often less; a value of m indicates that the relative error is likely to be less than 10^{-m}.

If the final value of a sum is considerably smaller than some of the intermediate values, the associated K2 will be somewhat smaller than

would otherwise be the case. Cancellation will have taken place, and this will usually mean reduced accuracy, but by "remembering" the larger value, it is possible to assess how damaging the cancellation has been.

There are three short subprograms used to monitor the convergence procedure. IBIG has two arguments, A and N, and its purpose is a magnitude comparison. N is thought of as a "descriptive integer" so that its high part represents a power of 2. If this power of 2 is as large as the magnitude of A, then IBIG(A,N) is given the value N; otherwise M is determined to satisfy $2^{M-1} \leq |A| < 2^{M}$ and the high part of IBIG(A,N) is set to 128(M + 128) with the low part (7 least significant bits) taken from N. TWØK has the single argument K, and TWØK(K) is the floating-point number 2^{M} if the high part of K is 128(M + 128). LLØW also has the single argument K, and LLØW(K) is just the integer given by the low order 7 bits of K. These subprograms are given in FØRTRAN, but equivalent assembly language programs are easily written and operate much more quickly.

Aside from this convergence procedure, most of the coding is fairly straightforward and not strikingly novel. Perhaps the most unusual feature is the Bessel function computation, and even here the method has become almost standard. The program requires $J_0(z)$, $J_1(z)$,..., for a given (possibly complex) argument z; there is also a need for $Y_0(z)$, $Y_1(z)$,..., and for some scaled Bessel functions that we refer to as $\tilde{J}_n$ and $\tilde{Y}_n$, but each Y sequence is easily obtained from the corresponding J sequence and the $\tilde{J}_n$'s are computed in virtually the same manner as the J_n's, so we discuss only the computation of J_0, J_1, etc., for a given z. The basic tool used is the recurrence formula,

$$J_{n+1}(z) = \frac{2n}{z} J_n(z) - J_{n-1}(z).$$

For a fixed z, this formula can, in principle, be used to generate $J_{n+1}(z)$ for $n \geq 1$, if $J_0(z)$ and $J_1(z)$ are known. But the J sequence decreases very rapidly with increasing n, while the complementary Y sequence increases very rapidly. Unless the values taken for J_0 and J_1 are exact and all subsequent calculation is also exact, the computed J sequence will contain an initially small, but eventually dominant component of the Y sequence. For typical values of z and good single-precision approximations of J_0 and J_1, we often find a computed J_{10} to be worthless.

It is possible however, to turn this behavior to advantage; if J_0, J_1,...J_n are desired, one selects a suitable $m > n$ and defines $F_m = 0$,

$F_{m-1} = \epsilon$ with ϵ small but arbitrary. The recurrence formula is used in reverse order to generate the sequence $F_{m-2}, F_{m-3}, \ldots, F_1, F_0$. The F sequence then satisfies $F_k = \alpha J_k + \beta Y_k$ for some α and β independent of k; in particular, we have $F_m = 0 = \alpha J_m + \beta Y_m$, so that $\frac{\beta}{\alpha} = -\frac{J_m}{Y_m}$ and we can then write $F_k \sim \alpha J_k$ with relative error satisfying

$$\left| \frac{F_k - \alpha J_k}{\alpha J_k} \right| = \left| \frac{\beta Y_k}{\alpha J_k} \right| = \left| \frac{J_m}{Y_m} \right| \cdot \left| \frac{Y_k}{J_k} \right| = \left| \frac{J_m}{J_k} \right| \cdot \left| \frac{Y_k}{Y_m} \right| .$$

If m is considerably larger than k, we will have $|J_m| << |J_k|$ and also $|Y_k| << |Y_m|$ so that the relative error is the product of two small factors, and it is reasonable to assume $F_k = \alpha J_k$, at least for computational purposes. In practice it is sufficient to take m equal to n+30 to produce good single precision accuracy for fairly small z, say $|z| \leq 50$. To complete calculation of the J's, we need only determine the value of α. This is done by using the relation:

$$J_0(z) + 2J_2(z) + 2J_4(z) + \cdots\cdots = 1$$

in the form

$$F_0(z) + 2F_2(z) + 2F_4(z) + \cdots\cdots = (\alpha)(1) = \alpha .$$

If convergence of this series is overly slow - and this happens very rarely, if at all - then we substitute a direct calculation of $J_0(z)$ to yield $\alpha = \frac{F_0}{J_0}$. This algorithm is somewhat prone to overflow, so provision is made for rescaling the F sequence if necessary. If z is complex, then α is ordinarily complex and caution must be exercized in dividing by α. In the event α (= a + ib) is large, one should avoid forming $a^2 + b^2$, which may easily overflow when a and b do not. Recurrence techniques for generating Bessel functions have been discussed by Goldstein and Thaler.*

Another problem of computational interest is the solution of a set of linear equations. The subroutine LINSYS has been used to accomplish this task. It employs a direct Gaussian elimination scheme, with so-called partial pivoting. This means that at the k^{th} stage, the pivot

---

* Goldstein, M. and Thaler, R. M. "Recurrence Techniques for the Calculation of Bessel Functions", Mathematical Tables and Other Aids to Computation, Vol. 13, 65-68, 1959, pp. 102-108.

is chosen as the largest element in the k^{th} column, except that above-diagonal elements are not eligible for consideration. An interchange of rows then takes place, if necessary, to place the pivot in the diagonal position of column k. We then subtract multiples of row k from the subsequent rows, so as to introduce zeros into the lower part of column k. After n-1 steps of the algorithm, the original system of equations has been replaced by an upper triangular system whose solution presents no special difficulties.

The roles of the subprograms IBIG, TWØK, LLØW, and LINØYØ have already been described. For convenience, we list the remaining subroutines and briefly indicate their functions:

BESSEL calculates the sequence $J_0(z), J_1(z), \ldots J_n(z)$ for real or complex z.

TILDE computes the scaled Bessel functions $\tilde{J}_0(z), \tilde{J}_1(z), \ldots, \tilde{J}_n(z)$; and also $\tilde{Y}_0(z), \tilde{Y}_1(z), \ldots, \tilde{Y}_n(z)$

These functions are defined in Vol. I, Appendix A, equations (A.13).

JTILØ is used by BESSEL AND TILDE to supply a value for $J_0(z)$.

ZRHSP and RHSP generate the right sides, M_n of the systems of equations for n = 0 and n >0, respectively, if the wave is dilatational.

ZRHSM AND RHSM generate these right sides for the shear wave.

Aside from these subroutines, the program employs only the standard library tape subprograms.

B. Input Specification

The first card has the format (A6, 17I3). It should contain a six-character code to identify the run, followed by 13 integers. The code may include any combination of the 10 decimal digits, the 26 letters, and the various special characters, including blanks. It is reproduced on the output but has no other effect. The 13 integers are, in order, KØDEN, KTYPE, KØDØP, KCØ, MAXØUT, KAPP, MØTAP, KAPN, MØTAN, NACRCY, NRØ , NTH, and NMAJ; they should appear in columns 7-9, 10-12, 13-15, etc. The significance of KAPP, MØTAP, KAPN, MØTAN, and NACRCY has already been indicated. KØDEN is taken as 0, 1, -1,

accordingly, as the run is to describe a simple dilatational wave, a simple shear wave, or a combined wave, respectively. The value KTYPE = 1 is used for a rectangular wave, KTYPE = 2 for a triangular wave, and KTYPE = 3 for an exponential decay wave. For a combined wave (KØDEN = -1), this value of KTYPE applies only to the dilatational part and a new value is to be furnished for the shear wave; we shall return to this consideration later. KØDØP should be given as 1 if a Lanczos factor (see Vol. I, Section VI. 2) is to be included, and as 0 otherwise. With KCØ >0, there is a large volume of intermediate output (see the sample output). If KCØ ≤ 0, the intermediate output is suppressed. Similarly, if MAXØUT ≠ 0, the maxima (of the stress quantities) with respect to ρ, θ, and $\bar{x}$ and the overall maxima are computed and printed. If MAXØUT = 0, this output is not given. NRØ is the number of ρ values to be used. NTH is the number of θ values explicitly given. NMAJ is the number of "major" $\bar{x}$ values given, but it is possible to insert equally spaced subdivision points between adjacent $\bar{x}$ values, as discussed below in connection with the array INTERX.

It should be noted that all the floating-point input is to be given in (E14.7) format. This applies in particular to the parameters E_1, $E_2, \ldots E_{11}$, and to the NRØ values of ρ, the NTH values of θ, and the NMAJ values of $\bar{x}$, which follow the first card in the order given. If NMAJ = 1, the next card is omitted, but otherwise there should then be a card containing NMAJ-1 integers in (20I3) format. These integers, called INTERX, indicate the number of intermediate $\bar{x}$ points to be inserted between the given, so-called "major" $\bar{x}$ values. Specifically, INTERX(I) stipulates the number of new points between XMAJ(I) and XMAJ(I+1).

If KTYPE = 1, this completes the input for a simple dilatational or shear wave, or for the dilatational part of a combined wave. If KTYPE = 2 or 3, two more (E14.7) cards are required, with values of CTBIN and CAYIN or EMIN, respectively. No additional input is needed for either of the simple waves, but if a combined wave is to be described, several cards follow to establish the desired relation between the two parts. First is an integer card containing LEADX, KEYX, and NTH values KEYTH. LEADX specifies the XMAJ value that is to become the first $\bar{x}$ for which the shear wave calculation is to be performed, i.e., XMAJ(LEADX) serves as this $\bar{x}$. Similarly, XMAJ(KEYX) designates $\bar{x}^*$, the nondimensional time delay between pulses. Next is an integer card containing the values of KTYPE and KCØ that are to apply to the shear wave calculation, and then three cards containing PHI, AMPFAC, and a new E11 value. If KTYPE = 2 or 3, two additional cards are needed, with the new valves of CTBIN and CAYIN or EMIN, respectively.

The correspondence between FORTRAN input symbols and the actual parameters used in the analysis follows:

$E(1) = {}^{h}/R$	Liner thickness to mean radius ratio
$E(2) = \frac{b}{c_{dm}T}$	Inverse of non-dimensional 1/2 period of the wave form
$E(3) = \nu$	Poisson's ratio for the liner
$E(4) = \nu_m$	Possion's ratio for the medium
$E(5) = E_m/E$	Medium to liner ratio of Young's Modulii
$E(6) = \gamma/\gamma_m$	Liner to medium ratio of mass density
$E(7) = \frac{b\,\Omega_2}{c_{dm}}$	Inverse of non-dimensional relaxation time for shear stress
$E(8) = \frac{\Omega_1}{\Omega_2}$	Stress relaxation time versus uniaxial strain relaxation time
$E(9) = {}^{\tau_1}/\Omega_1$	Stress relaxation time versus strain recovery time for uniaxial strain
$E(10) = \frac{\tau_2}{\Omega_2}$	Shear stress relaxation time versus shear strain relaxation time
$E(11) = \frac{c_{dm}t_o}{b}$	Nondimensional rest time
$PHI = \phi$	Angle between incoming dilatational and shear wave
$AMPHAC = \frac{\tau_o}{\sigma_o}$	Amplitude ratio, shear wave to dilatational wave

CAYIN = k | Integer, describing the decay time of triangular wave form; decay time = $k\left(\frac{c_m(t_1-t_0)}{b}\right)$

CTBIN = $\frac{c_m(t_1-t_0)}{b}$ | Non-dimensional rise time for triangular or linear rise-exponential decay wave form

$$c_m = \begin{cases} c_{dm} & \text{dilatational wave} \\ c_{tm} & \text{shear wave} \end{cases}$$

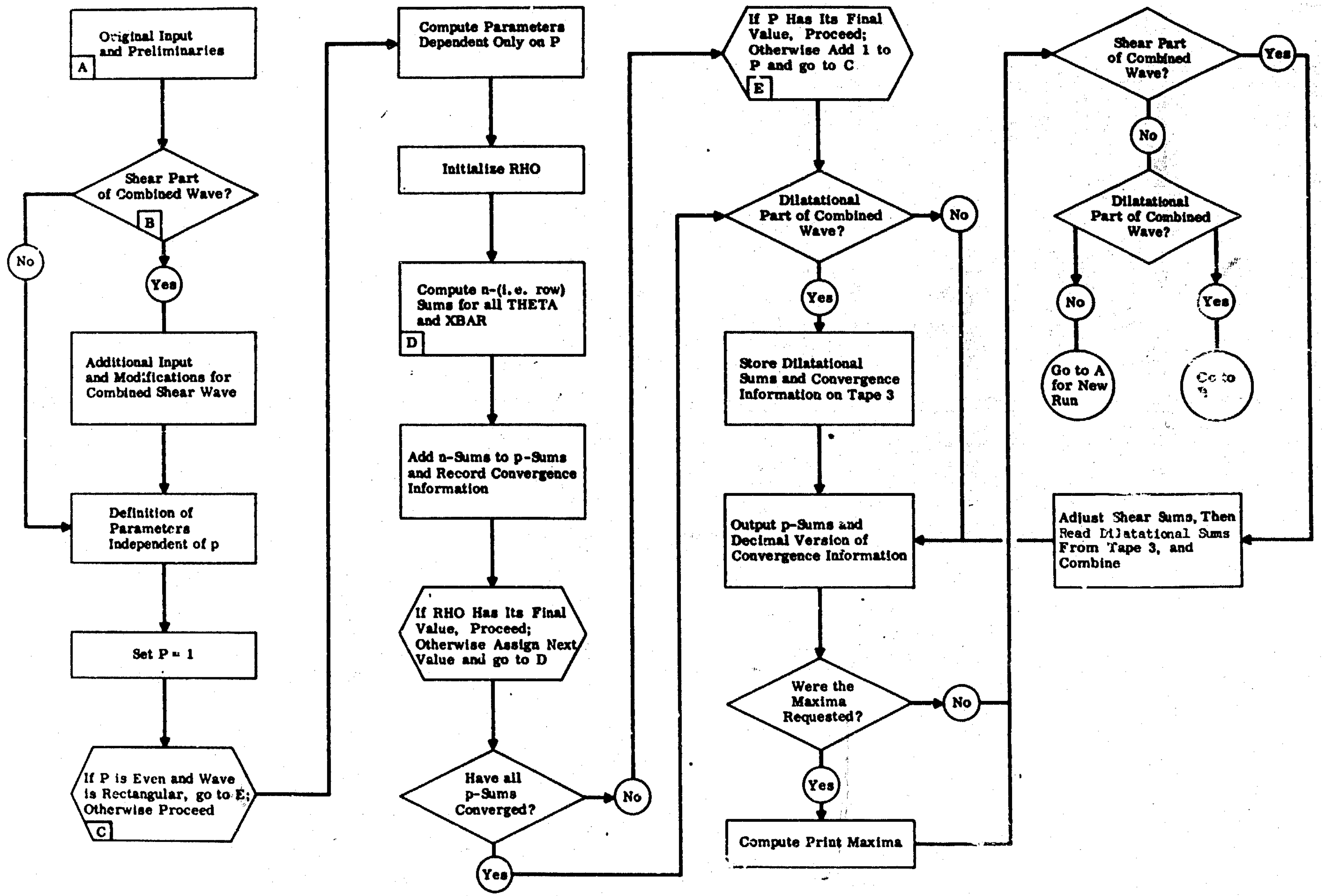
EMIN = k | Parameter associated with linear rise - exponential decay waveform and which defines the time at which the stress in the exponential decay portion of the pressure-time history is negligible, e.g.,

$$e^{-k} \ll 1.$$

C. Flow Charts

This subsection contains two flow charts, intended to assist in the understanding of the program's flow of control. Chart I is relatively brief; it attempts to convey a sense of the over-all structure of the procedure and keeps the use of special symbols to a minimum. More details are presented in Chart II.

CHART I



Original Input and Preliminaries
A
Shear Part of Combined Wave?
B
Yes
No
Additional Input and Modifications for Combined Shear Wave
Definition of Parameters Independent of p
Set P = 1
If P is Even and Wave is Rectangular, go to E; Otherwise Proceed
C
Compute Parameters Dependent Only on P
Initialize RHO
Compute n-(i.e. row) Sums for all THETA and XBAR
D
Add n-Sums to p-Sums and Record Convergence Information
If RHO Has Its Final Value, Proceed; Otherwise Assign Next Value and go to D
Have all p-Sums Converged?
No
Yes
If P Has Its Final Value, Proceed; Otherwise Add 1 to P and go to C
E
Dilatational Part of Combined Wave?
No
Yes
Store Dilatational Sums and Convergence Information on Tape 3
Output p-Sums and Decimal Version of Convergence Information
Were the Maxima Requested?
No
Yes
Compute Print Maxima
Shear Part of Combined Wave?
Yes
No
Dilatational Part of Combined Wave?
No
Yes
Go to A for New Run
Adjust Shear Sums, Then Read Dilatational Sums From Tape 3, and Combine

CHART II

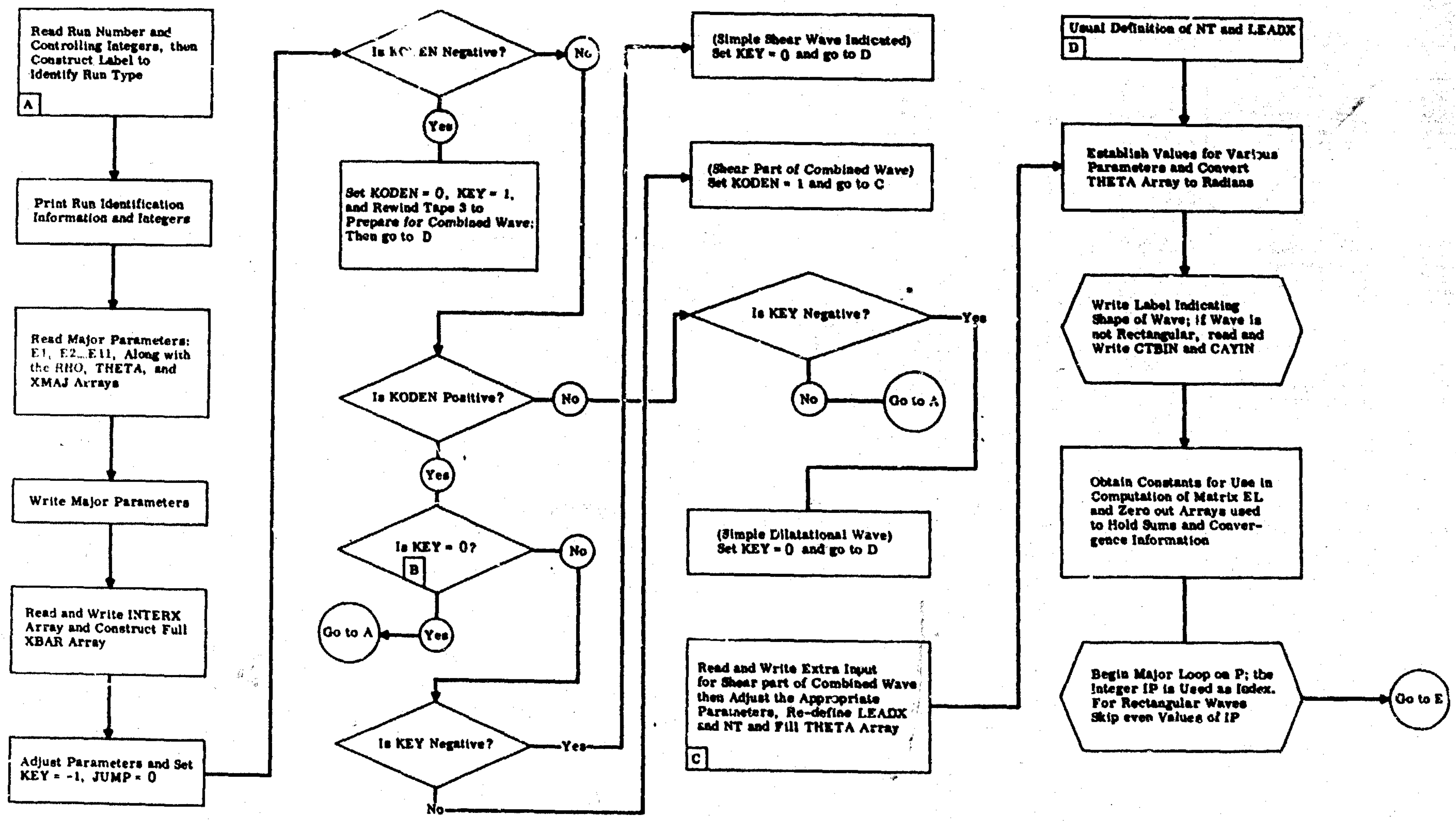

Read Run Number and Controlling Integers, then Construct Label to Identify Run Type
A
Print Run Identification Information and Integers
Read Major Parameters: E1, E2...E11, Along with the RHO, THETA, and XMAJ Arrays
Write Major Parameters
Read and Write INTERX Array and Construct Full XBAR Array
Adjust Parameters and Set KEY = -1, JUMP = 0
Is KODEN Negative?
No
Yes
Set KODEN = 0, KEY = 1, and Rewind Tape 3 to Prepare for Combined Wave; Then go to D
Is KODEN Positive?
No
Yes
Is KEY = 0?
B
No
Yes
Go to A
Is KEY Negative?
Yes
No
(Simple Shear Wave Indicated) Set KEY = 0 and go to D
(Shear Part of Combined Wave) Set KODEN = 1 and go to C
Is KEY Negative?
Yes
No
Go to A
(Simple Dilatational Wave) Set KEY = 0 and go to D
Read and Write Extra Input for Shear part of Combined Wave then Adjust the Appropriate Parameters, Re-define LEADX and NT and Fill THETA Array
C
Usual Definition of NT and LEADX
D
Establish Values for Various Parameters and Convert THETA Array to Radians
Write Label Indicating Shape of Wave; if Wave is not Rectangular, read and Write CTBIN and CAYIN
Obtain Constants for Use in Computation of Matrix EL and Zero out Arrays used to Hold Sums and Convergence Information
Begin Major Loop on P; the Integer IP is Used as Index. For Rectangular Waves Skip even Values of IP
Go to E

CHART II (CONTINUED)

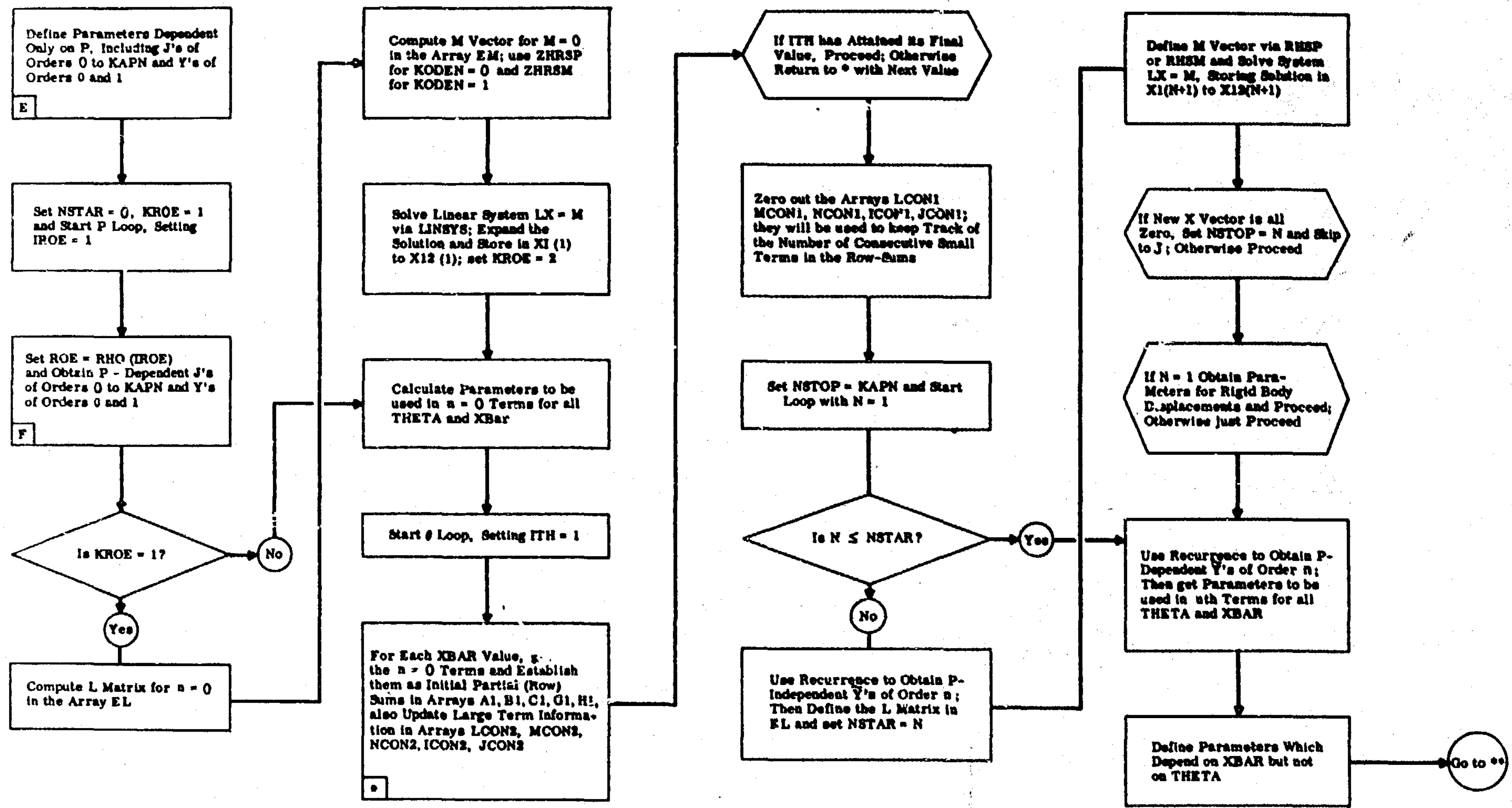


CHART II (CONTINUED)

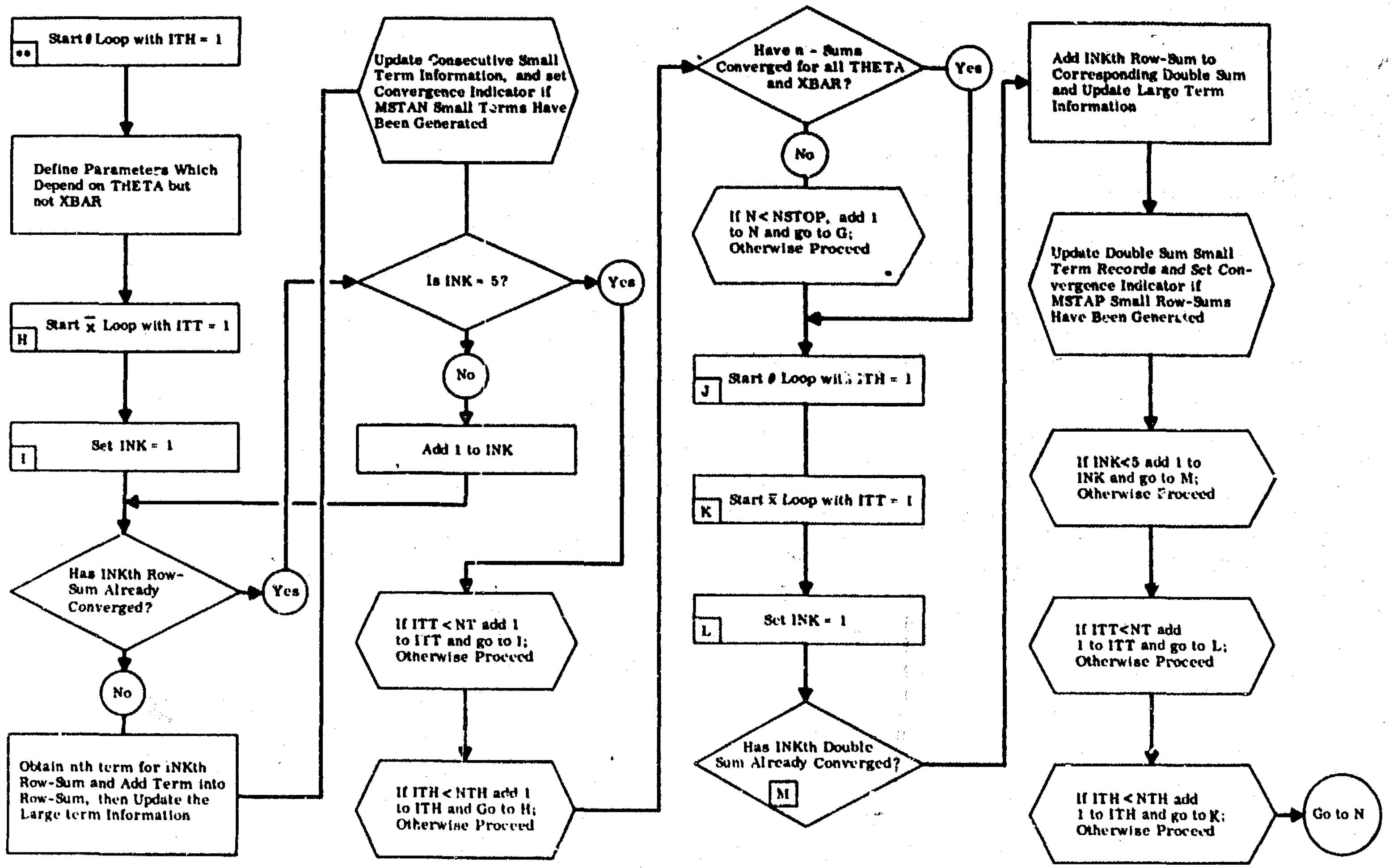
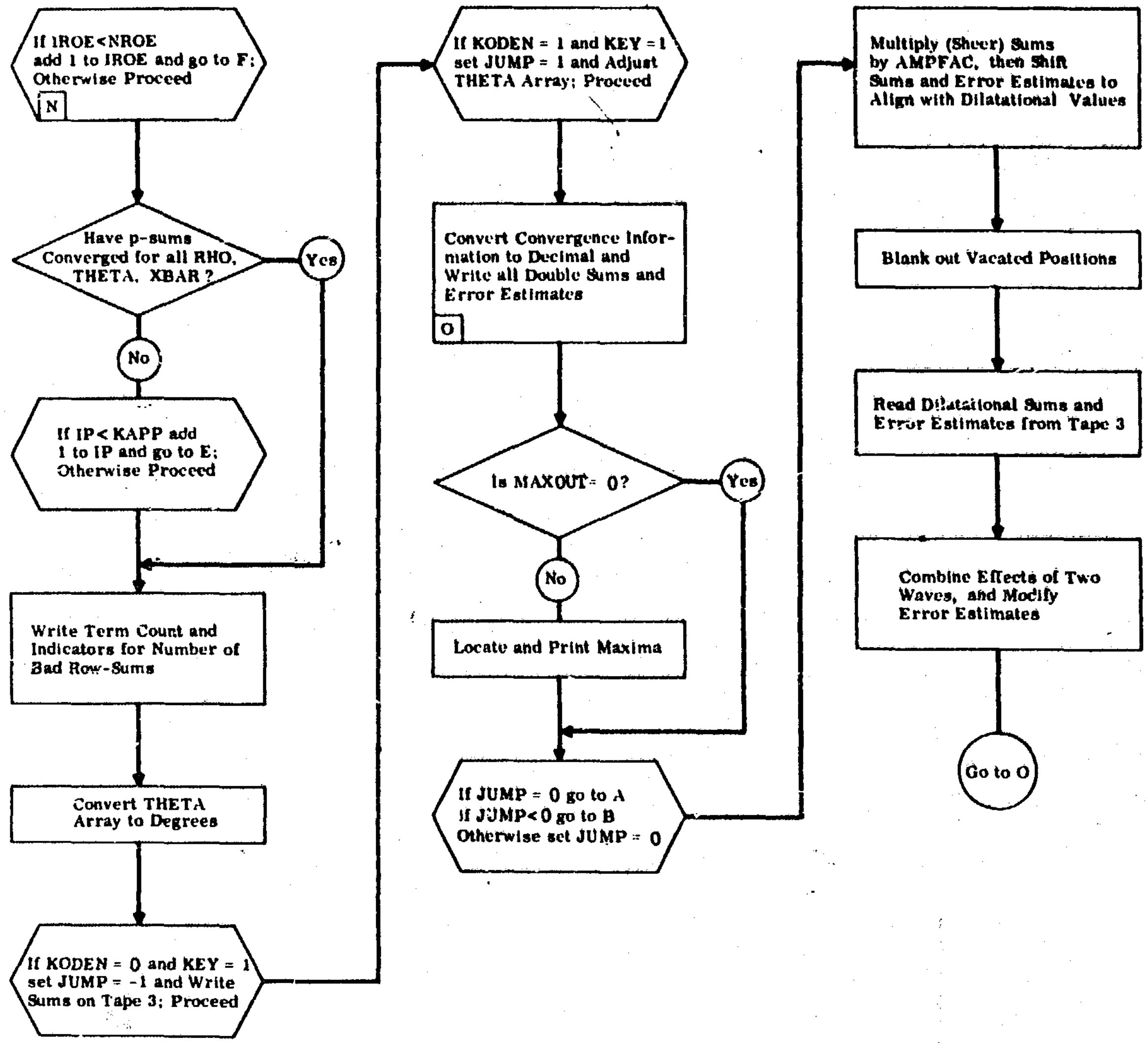


CHART II (CONTINUED)



SECTION II

DISCUSSION OF INPUT PARAMETERS AND EXAMPLES OF PREPARATION OF INPUT DATA CARDS

A. General Remarks

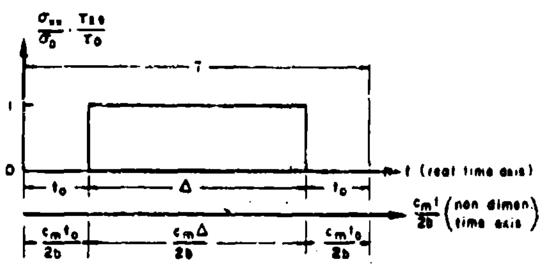
The efficient use of the computer program depends in part on the proper choice of some key parameters. These are KAPP, KAPN, MØTAP, MØTAN, NACRCY, $c_m t_o/b$, and $b/c_{dm}T$. In all the data presented in Vol I, MØTAP, MØTAN, NACRCY were taken as 4, 6, 4, respectively. These values were considered to be adequate for obtaining the plotting accuracy desired. Although increasing the value of these parameters should increase accuracy (and also increase the computer time), the exact limitations and effect of such an increase has not been studied.

The accuracy of any computation is also dependent on the choice of KAPP since KAPP controls the Fourier representation of the wave form. In general, KAPP was chosen as 101 for the rectangular incident wave form, 200 for the triangular wave form and greater than 300 for the linear rise exponential decay wave form. These values were adequate, in most cases, for obtaining approximately four figure accuracy for all the quantities of interest. KAPN was chosen as 50 for all the results presented in Vol I. However, this value is conservative and in most cases it can be decreased considerably. The determination of $b/c_{dm}T$ and $c_m t_o/b$ is discussed in the next subsection.

B. Calculation of Auxiliary Parameters

The significance of the parameters controlling "rest time" and the wave form were discussed in Sec. V.1 and Appendix B of Vol. I. In the present subsection, the procedure for determining the value of these parameters is discussed for each wave form. It is assumed that material and geometric properties of the liner and medium are given (i.e., c_{dm}, c_{tm}, b) and that only non-dimensional parameters are discussed. To facilitate the discussion, components of Fig. 2b, Vol I, showing the various wave forms and associated parameters, are reproduced immediately prior to the discussion of the wave form being considered.

RECTANGULAR WAVE FORM



$$c_m = \begin{cases} c_{dm} & \text{for incoming dilatational wave} \\ c_{tm} & \text{for incoming shear wave.} \end{cases}$$

As mentioned in Appendix B, Vol I, any two of the parameters $b/c_{dm}T$, $c_m t_o/b$, $c_m \Delta/b$ may be chosen arbitrarily. In preparing input data for the reported problems it was most convenient to choose $c_m t_o/b$ and $c_m \Delta/b$, the dimensionless "rest time" and pulse length, respectively, and then solve for $b/c_{dm}T$ from

$$\frac{b}{c_{dm}T} = \frac{c_m/c_{dm}}{2\left(\frac{c_m t_o}{b}\right) + \left(\frac{c_m \Delta}{b}\right)} \tag{1}$$

which is obtained from Eq. (5.[illegible]) of Appendix B, Vol I.

To demonstrate this computation for a dilatational incident wave, we choose

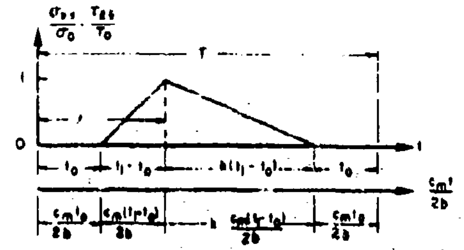
$$\frac{c_{dm} t_o}{b} = 20$$

$$\frac{c_{dm} \Delta}{b} = 20,$$

and find from Eq. 1, that $b/c_{dm}T$ 0.01666667.

For most of the problems that were studied, $2c_m t_o/b \approx 40$ (with $20 \leq c_{dm}T/b \leq 200$) was sufficient to ensure rest conditions before the pulse strikes the cylinder.

TRIANGULAR WAVE FORM



c_m : $\begin{cases} c_{dm} & \text{for incoming dilatational wave} \\ c_{sm} & \text{for incoming shear wave} \end{cases}$

The parameters that define the wave form and rest time are $c_m(t_1-t_o)/b$, k, $b/c_{dm}T$, and $c_m t_o/b$. As discussed in Appendix B, Vol. I, only three of these parameters can be chosen arbitrarily. Since it is natural that $c_m(t_1-t_o)/b$ and k are prescribed, we are free to choose either

a) $b/c_{dm}T$, and calculate $c_m t_o/b$, or

b) $c_m t_o/b$, and calculate $b/c_{dm}T$.

Again, for the triangular wave form it was most convenient to choose $c_m t_o/b$ along with k and $c_m(t_1-t_o)/b$, and find $b/c_{dm}T$ from

$$\frac{b}{c_{dm}T} = \frac{c_m/c_{dm}}{\frac{2c_m t_o}{b} + (k+1)\frac{c_m(t_1-t_o)}{b}}, \tag{2}$$

where Eq. 2 is obtained from Eq. (B.6) Appendix B, Vol I.

Thus, for a dilatational incident wave, given

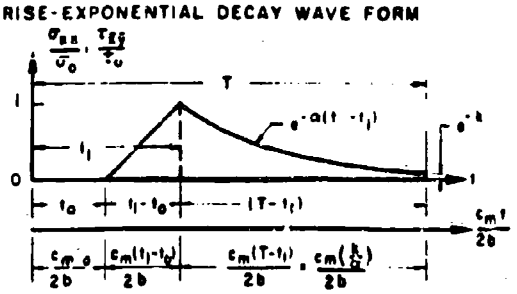
$c_m(t_1-t_o)/b = 1$ rise time

$k = 5$ and decay time of pulse,

we choose $\frac{2c_m t_o}{b} = 40$,

and find from Eq. 2 $\frac{b}{c_{dm}T} = 0.021739$.

LINEAR RISE-EXPONENTIAL DECAY WAVE FORM



$c_m = \begin{cases} c_{dm} & \text{for incoming dilatational wave} \\ c_{tm} & \text{for incoming shear wave} \end{cases}$

Note: k is selected so that the incident stress is negligeably small at t=T

The parameters defining the wave form and rest time are $b/c_{dm}T$, $c_m t_o/b$, $c_m(k/\alpha)/b$, $c_m(t_1-t_o)/b$. Again, we are free to select only three of these parameters arbitrarily. Since $c_m(k/\alpha)/b$ and $c_m(t_1-t_o)/b$ are always given, we can select either $b/c_{dm}T$ or $c_m t_o/b$ arbitrarily.

As for the other two wave forms, it is again most convenient to choose $c_m t_o/b$ and calculate $b/c_{dm}T$. Thus, from this wave form,

$$\frac{b}{c_{dm}T} = \frac{c_m}{c_{dm}} \left[\frac{1}{\frac{c_m t_o}{b} + \frac{c_m(t_1-t_o)}{b} + \frac{c_m(k/\alpha)}{b}} \right]$$

Hence, for an incident dilatational wave, given

$$\frac{c_{dm}(1/\alpha)}{b} = 5 \quad \text{and} \quad \frac{c_{dm}(t_1-t_o)}{b} =$$

we take $k = 12$ and $\frac{c_{dm}t_o}{b} = 30$

and calculate $b/c_{dm}T = 0.010989$.

C. **Examples**

Example 1 - Preparation of input for a dilatational rectangular wave followed by a rectangular shear wave.

Given geometric and material properties are:

Slow Granite Medium (Elastic)

$E_m = 1.0 \times 10^6$ psi

$\nu_m = .25$

$\gamma_m = 5.2$ slug/ft^3

$$\frac{\Omega_1}{\Omega_2} = \frac{\tau_1}{\Omega_1} = \frac{\tau_2}{\Omega_2} = 1.0$$

$$\frac{b\,\Omega_2}{c_{dm}} = 0.0$$

Concrete Liner

$E = 2.5 \times 10^6$ psi

$\nu = .2$

$\gamma = 4.5$ slug/ft^3

Thickness $h = 2.381$ ft.

Mean Radius $R = 23.81$ ft.

or

Outer Radius $b = 25$ ft.

Inner Radius $a = 22.62$ ft.

Calculated quantities are:

$$\frac{h}{R} = .1$$

$$\frac{E_m}{E} = .4$$

$$\frac{\gamma}{\gamma_m} = .86538462$$

$$c_{dm} = \sqrt{\frac{E_m\,(1 - \nu_m)}{\gamma_m\,(1 - \nu_m)\,(1 + \nu_m)}} = 5765 \text{ ft/sec.}$$

$$c_{tm} = \sqrt{\frac{E_m}{2(1 + \nu_m)\,\gamma_m}} = 3328$$

If the duration of the rectangular dilatational wave is,

$\Delta_d = .08673$ sec.,

then

$$\frac{c_{dm}\,\Delta_d}{b} = 20$$

Choosing $\dfrac{c_{dm}t_o}{b} = 20,$

we find

$$\frac{b}{c_{dm}T} = \frac{1}{2\,\dfrac{c_{dm}t_o}{b} + \dfrac{c_{dm}\,\Delta_d}{b}} = \frac{1}{60} = .01666667.$$

The input data, as shown, coupled with the selection of ρ, θ, $\overline{x}$ points, can be used for a dilatational run alone (changing KØDEN = -1 to 0).

Additional information required for the shear wave is:

To maintain generality, the duration of the sheer wave is assumed to be

$$\Delta_s = .04336 \text{ sec.}$$

Therefore,

$$\frac{c_{dm}\,\Delta_s}{b} = 10,$$

and we calculate

$$\frac{c_{tm}t_o}{b} = \frac{1}{2}\left(\frac{c_{tm}}{c_{dm}}\right)\left(\frac{1}{b/c_{dm}T} - \frac{c_{dm}\,\Delta_s}{b}\right) = 14.425.$$

For an air induced loading, traveling along the ground at the super-seismic velocity, U = 8500 ft/sec, the angle between the dilatational and shear wave is found to be $\phi = 19.7^{\circ}$. If it is assumed that each wave travels approximately 150 ft before it strikes the cylinder, then the shear wave strikes the cylinder 19 mil sec after the dilatational wave. The non-dimensional time delay is therefore found to be

$$\frac{c_{dm}\,\delta t}{b} = 4.381.$$

On empirical grounds, the amplitude ratio, $\frac{\phi_o}{\Phi_o}$, of the two incident waves is taken to be 1/3. To complete the input information, ρ, θ, $\bar{x}$ points are chosen.

For the demonstration problem, five θ points were chosen for the dilatational part. Two extra θ points are selected for the shear and combination part [specified by KEYTH(I)]. Five major stations for $\bar{x}$ were chosen; one point (-1.0) before the wave strikes the cylinder, to check that all stress and displacement quantities are approximately zero, and a point, $\bar{x} = 4.381$, corresponding to the time the shear wave strikes the cylinder. It should be mentioned that this latter point is required information for all combination runs. An additional 24 points are also included [controlled by INTERX(I)]. Only one ρ point is used in this case, $\rho = 0.9047619$, the inner surface.

A chart with all input data for this example, is shown on the next two pages.

EXAMPLE - INPUT DATA

INTEGER CARD

(1)

RUN NO.		MODEN	KTYPE	KODSP	KCO	MAXOUT	KAPP	MSTAP	KAPN	MSTAN	NACRCY	NRO	NTN
3	6	9	12	15	18	21	24	27	30	33	36	39	42
IC	ON	-1	1	1	-1	0	151	4	50	6	4	1	5

NMAJ
45
6

(2)

$\frac{b}{R}$	0.1
$\frac{b}{c_{dm}\gamma}$	0.016666667
ν	0.2
ν_m	0.25
$\frac{\epsilon_m}{\epsilon}$	0.4
$\frac{\gamma}{\gamma_m}$	0.86538462
$\frac{b}{\Omega_2 c_{dm}}$	0.0
$\frac{\Omega_1}{\Omega_2}$	1.0
$\frac{r_1}{\Omega_1}$	1.0
$\frac{r_2}{\Omega_2}$	1.0
$\frac{c_m t_o}{b}$	20.0

(3)

ρ_1	0.9047619
ρ_2	
ρ_3	

MAXIMUM NUMBER OF ρ, POINTS IS 3

(4)

θ_1	0.0
θ_2	64.7
θ_3	90.0
θ_4	154.7
θ_5	180.0
θ_6	
θ_7	
θ_8	
θ_9	
θ_{10}	

MAXIMUM TOTAL NUMBER OF θ POINTS IS 10

(5)

$\bar{x}_1$	- 1.0
$\bar{x}_2$	4.0
$\bar{x}_3$	4.381
$\bar{x}_4$	4.5
$\bar{x}_5$	10.5
$\bar{x}_6$	16.0
$\bar{x}_7$	
$\bar{x}_8$	
$\bar{x}_9$	
$\bar{x}_{10}$	

MAXIMUM NUMBER OF $\bar{x}_{maj}$ POINTS IS 10

(6)

1	2	3	4	5	6	7	8	9
9	0	0	11	4				

INTERX(I) DELETE IF NMAJ=1

(7)

ADDITIONAL INPUT FOR KTYPE = 2,3

$\frac{c_m(t_1-t_0)}{b}$	
k	

EXAMPLE 1 INPUT DATA (Cont)

ADDITIONAL INPUT FOR A SHEAR WAVE SUPERPOSED WITH DILATATIONAL WAVE

(1)

LEADX			KEYX			KEYTH (1): 1			2			3			4			5		
		2			3			0			1			0			1			0

(2)

KTYPE			KCO		
		1		-	1

(3)

ϕ	19.7
$\frac{\tau_o}{\sigma_o}$	0.33333333
$\frac{c_m t_o}{b}$	14.425

ADDITIONAL INPUT IF KTYPE = 2,3

(4)

$\frac{c_m(t_1-t_b)}{b}$	
k	

Example 2 - Preparation of input for an incident rectangular shear wave.

The preparation of input for this case follows from the discussion of a dilatational rectangular wave (dilatational case, Example 1) with the following changes.

1) KODEN = 1

2) $\frac{c_{dm} t_o}{b} \longrightarrow \frac{c_{tm} t_o}{b}$

3) $$\frac{b}{c_{dm} T} = \frac{c_{tm}/c_{dm}}{2\left(\frac{c_{tm} t_o}{b}\right) + \left(\frac{c_{tm} \Delta}{b}\right)}$$

Accordingly, choosing

$$\frac{c_{tm} t_o}{b} = 20, \quad \text{with} \quad \frac{c_{tm} \Delta}{b} = 10,$$

we find

$$\frac{b}{c_{dm} T} = .01155$$

A chart with all input data for this problem is on the next page.

EXAMPLE 2 INPUT DATA

RUN NO.	KODEN	KTYPE	KODSP	KCO	MAX OUT	KAPP	MSTAP	KAPN	MSTAN	NACRCY	NRO	NTH
3 6	9	12	15	18	21	24	27	30	33	36	39	42
26	1	1	1	-1	1	101	4	50	6	4	1	5

(1)

NMAJ
45
4

INTEGER CARD

(2)

$\frac{h}{R}$	0.1
$\frac{b}{c_{dm}T}$	0.01155
ν	0.2
ν_m	0.25
$\frac{E_m}{E}$	0.4
$\frac{\gamma}{\gamma_m}$	0.86538462
$\frac{b}{\Omega_2 e_{dm}}$	0.0
$\frac{\Omega_1}{\Omega_2}$	1.0
$\frac{\tau_1}{\Omega_1}$	1.0
$\frac{\tau_2}{\Omega_2}$	1.0
$\frac{e_m \tau_0}{b}$	20.0

(3)

ρ_1	0.9047619
ρ_2	
ρ_3	

MAXIMUM NUMBER OF ρ. POINTS IS 3

(4)

θ_1	0.0
θ_2	45.0
θ_3	90.0
θ_4	135.0
θ_5	180.0
θ_6	
θ_7	
θ_8	
θ_9	
θ_{10}	

MAXIMUM TOTAL NUMBER OF θ POINTS IS 10

(5)

$\bar{x}_1$	-2.0
$\bar{x}_2$	0.0
$\bar{x}_3$	5.0
$\bar{x}_4$	15.0
$\bar{x}_5$	
$\bar{x}_6$	
$\bar{x}_7$	
$\bar{x}_8$	
$\bar{x}_9$	
$\bar{x}_{10}$	

MAXIMUM NUMBER OF $\bar{x}_{maj}$ POINTS IS 10

(6)

1	2	3	4	5	6	7	8	9
0	9	9						

INTERX(I) DELETE IF NMAJ=1

(7)

ADDITIONAL INPUT FOR KTYPE = 2,3

$\frac{e_m(T_1-T_0)}{b}$	
h	

Example 3 - Preparation of input for an incident triangular shear wave (KTYPE = 2).

Medium: Porous Sandstone (Elastic)

$E_m = 3.0 \times 10^6$ psi

$\nu_m = 0.2$

$\gamma_m = 4.44$ slug/ft^3

$\frac{E_m}{E} = 1.2$

$\frac{\gamma}{\gamma_m} = 1.0135135$

Liner: Concrete

$E = 2.5 \times 10^6$ psi

$\nu = 0.2$

$\gamma = 4.5$ slug/ft^3

$\frac{h}{R} = 0.1$

$b = 25$ ft.

Parameters describing the wave form:

Rise time = 10 mil sec

Decay time = 50 mil sec

In non-dimensional form the parameters describing the triangular wave form are

$$c_{tm}(t_1-t_o)/b = 2.5468$$

$$k = 5$$

Choosing the rest time as

$$c_{tm}t_o/b = 30$$

we calculate $b/c_{dm}T$ from

$$\frac{b}{c_{dm}T} = \frac{c_{tm}/c_{dm}}{\frac{2c_{tm}t_o}{b} + (k+1)\frac{c_{tm}(t_1-t_o)}{b}} = .0081389.$$

To complete the information, we choose the ρ, θ, $\bar{x}$ points at which calculations are desired. In this example, three θ points, one ρ point and two $\bar{x}$ major points are chosen.

The input for a dilatational incident triangular wave form follows from this discussion with the indicated changes:

1) KØDEN = 0

2) $c_{tm}t_o/b \longrightarrow c_{dm}t_o/b$

$c_{tm}(t_1-t_o)/b \longrightarrow c_{dm}(t_1-t_o)/b$

3) $$b/c_{dm}T = \frac{1}{\frac{2c_{dm}t_o}{b} + (k+1)\frac{c_{dm}(t_1-t_o)}{b}}$$

The following chart contains the input data for this problem:

EXAMPLE 3 INPUT DATA

(1)

RUN NO.	KODEN	KTYPE	KODSP	KCO	MAX CUT	KAPP	MSTAP	KAPN	MSTAN	NACRCY	NRO	NTH
6	9	12	15	18	21	24	27	30	33	36	39	42
3A	1	2	0	-1	0	201	4	50	6	4	1	3

NMAJ
45
2

INTEGER CARD

(2)

$\frac{h}{R}$	0.1
$\frac{b}{c_{dm}T}$	0.0081389
ν	0.2
ν_m	0.2
$\frac{E_m}{E}$	1.2
$\frac{\gamma}{\gamma_m}$	1.0135135
$\frac{b}{\Omega_2 c_{dm}}$	0.0
$\frac{\Omega_1}{\Omega_2}$	1.0
$\frac{\tau_1}{\Omega_1}$	1.0
$\frac{\tau_2}{\Omega_2}$	1.0
$\frac{c_m t_0}{b}$	30.0

(3)

ρ_1	.9047619
ρ_2	
ρ_3	

MAXIMUM NUMBER OF ρ, POINTS IS 3

(4)

θ_1	0.0
θ_2	45.0
θ_3	135.0
θ_4	
θ_5	
θ_6	
θ_7	
θ_8	
θ_9	
θ_{10}	

MAXIMUM TOTAL NUMBER OF θ POINTS IS 10

(5)

$\bar{x}_1$	-1.0
$\bar{x}_2$	7.0
$\bar{x}_3$	
$\bar{x}_4$	
$\bar{x}_5$	
$\bar{x}_6$	
$\bar{x}_7$	
$\bar{x}_8$	
$\bar{x}_9$	
$\bar{x}_{10}$	

MAXIMUM NUMBER OF $\bar{x}_{maj}$ POINTS IS 10

(6)

1	2	3	4	5	6	7	8	9
15								

INTERX(I) DELETE IF NMAJ=1

(7)

ADDITIONAL INPUT FOR KTYPE = 2,3

$\frac{c_m(t_1-t_0)}{b}$	2.5468
h	5.0

Example 4 - Preparation of input for an incident dilatational linear-rise-exponential decay wave form (KTYPE = 3).

The geometric and material properties are the same as Example 3. Parameters describing the wave form are:

$$\text{Rise time} = 10 \text{ mil sec}$$

$$\text{decay time constant} = \frac{1}{\alpha} \quad 50 \text{ mil sec}$$

Therefore,

$$c_{dm}(t_1 - t_0) / b = 4.1554,$$

$$c_{dm}(1/\alpha) / b = 20.78,$$

and choosing

$$c_{dm} t_o / b = 30,$$

$$k = 10,$$

$$\frac{b}{c_{dm}T} = \frac{1}{\dfrac{c_{dm}t_o}{b} + \dfrac{c_{dm}(t_1 - t_0)}{b} + \dfrac{c_{dm}(k/\alpha)}{b}} = .0041326.$$

It should be noted that since this wave form is the most difficult to represent, KAPP must be taken greater than 301 in order to ensure proper convergence.

The incident shear wave case is handled in an analogous manner, with the following exceptions:

1) KØDEN = 1

2) $c_{dm}t_o/b \longrightarrow c_{tm}t_o/b$

$c_{dm}(t_1 - t_o)/b \longrightarrow c_{tm}(t_1 - t_o)/b$

3) $b/c_{dm}T$ is calculated from

$$b/c_{dm}T = \frac{c_{tm}/c_{dm}}{\dfrac{c_{tm}t_o}{b} + \dfrac{c_{tm}(t_1 - t_o)}{b} + \dfrac{c_{tm}(k/\alpha)}{b}}$$

The input data for this problem are given in the chart on the following page.

EXAMPLE 4 INPUT DATA

(1)

RUN NO.	KODEN	KTYPE	KODSP	KCO	MAX OUT	KAPP	MSTAP	KAPN	MSTAN	NACRCY	NRO	NTN
3 6	9	12	15	18	21	24	27	30	33	36	39	42
4A	0	1	0	-1	0	301	4	50	6	4	1	3

NMAJ
45
2

INTEGER CARD

(2)

$\frac{h}{R}$	0.1
$\frac{b}{c_{dm}\gamma}$	0.0041326
ν	0.2
ν_m	0.2
$\frac{E_m}{E}$	1.2
$\frac{\gamma}{\gamma_m}$	1.0135135
$\frac{b}{\Omega_2 c_{dm}}$	0.0
$\frac{\Omega_1}{\Omega_2}$	1.0
$\frac{\tau_1}{\Omega_1}$	1.0
$\frac{\tau_2}{\Omega_2}$	1.0
$\frac{c_m t_0}{b}$	30.0

(3)

ρ_1	.9047619
ρ_2	
ρ_3	

MAXIMUM NUMBER OF ρ, POINTS IS 3

(4)

θ_1	0.0
θ_2	90.0
θ_3	135.0
θ_4	
θ_5	
θ_6	
θ_7	
θ_8	
θ_9	
θ_{10}	

MAXIMUM TOTAL NUMBER OF θ POINTS IS 10

(5)

$\bar{x}_1$	-1.0
$\bar{x}_2$	7.0
$\bar{x}_3$	
$\bar{x}_4$	
$\bar{x}_5$	
$\bar{x}_6$	
$\bar{x}_7$	
$\bar{x}_8$	
$\bar{x}_9$	
$\bar{x}_{10}$	

MAXIMUM NUMBER OF $\bar{X}_{mej}$ POINTS IS 10

(6)

1	2	3	4	5	6	7	8	9
15								

INTERX(I) DELETE IF NMAJ=1

(7)

ADDITIONAL INPUT FOR KTYPE = 2,3

$\frac{c_m(t_1-t_0)}{b}$	4.1564
k	10.0

Example 5 - Preparation of input for a rectangular wave when the medium is viscoelastic.

The preparation of input for a viscoelastic medium follows formally from the elastic cases previously discussed. That is, while $\frac{b\Omega_2}{c_{dm}}$, $\frac{\Omega_1}{\Omega_2}$, $\frac{\tau_1}{\Omega_1}$, and $\frac{\tau_2}{\Omega_2}$ must be chosen appropriately, we choose the other input as previously discussed (using the same values for the other parameters, the previous examples in which

$$\frac{c_{dm}\,\Omega_2}{b} = 0, \quad \frac{\Omega_1}{\Omega_2} = \frac{\tau_1}{\Omega_1} = \frac{\tau_2}{\Omega_2} = 1.0$$

describe the elastic unrelaxed case).

Partly because viscoelasticity slows down the process of strain recovery in the particular medium chosen, it is much more difficult to pick rest times that are adequate to ensure that the initial conditions are satisfied at the time of arrival of the pulse. However, this difficulty can be surmounted in most cases by constructing each viscoelastic problem from two or more individual computer runs. For the first run the duration of the pulse is made small, thus increasing the allowable rest time and also decreasing the amount of creep deformation in the medium. A subsequent run can be one in which the desired pulse length or any intermediate length is used. For each subsequent run the initial conditions may not be satisfied, but at later times the results should be accurate enough to match those of the previous run. Ultimately, the response over the desired pulse duration can be constructed in this way.

An example set of data for a case for which this procedure was used is provided at the end of this discussion. It should be noted that the effects of viscoelasticity on the satisfaction of initial conditions as discussed above were not severe for the triangular and linear rise exponential decay wave forms that were studied.

The procedure outlined above does not apply only to viscoelastic cases. It should also be used for an elastic case whenever difficulty is encountered in attaining sufficient rest time and when more refined results are desired at early transit times.

Example input charts for a step pulse, in a viscoelastic medium involving two computer runs, are given on the following pages:

EXAMPLE 5 INPUT DATA

(1)

RUN NO.	KODEN	KTYPE	KODSP	KCO	MAX OUT	KAPP	MSTAP	KAPN	MSTAN	NACRCY	NRO	NTH
3 6	9	12	15	18	21	24	27	30	33	36	39	42
1 6 1 A	0	1	1	-1	0 1	0 1	4	5 0	6	4	1	3

NMAJ
45
3

INTEGER CARD

(2)

$\frac{h}{R}$	0.1
$\frac{h}{c_{dm}T}$	0.009090909
ν	0.2
ν_m	0.2
$\frac{E_m}{E}$	1.2
$\frac{\gamma}{\gamma_m}$	1.0135135
$\frac{b}{\Omega_2 c_{dm}}$	1.0
$\frac{\Omega_1}{\Omega_2}$	1.0
$\frac{\tau_1}{\Omega_1}$	0.1
$\frac{\tau_2}{\Omega_2}$	0.1
$\frac{c_m t_0}{b}$	44.0

(3)

ρ_1	0.9047619
ρ_2	
ρ_3	

MAXIMUM NUMBER OF ρ_1 POINTS IS 3

(4)

θ_1	0.0
θ_2	90.0
θ_3	180.0
θ_4	
θ_5	
θ_6	
θ_7	
θ_8	
θ_9	
θ_{10}	

MAXIMUM TOTAL NUMBER OF θ POINTS IS 10

(5)

$\bar{x}_1$	-1.0
$\bar{x}_2$	0.0
$\bar{x}_3$	21.0
$\bar{x}_4$	
$\bar{x}_5$	
$\bar{x}_6$	
$\bar{x}_7$	
$\bar{x}_8$	
$\bar{x}_9$	
$\bar{x}_{10}$	

MAXIMUM NUMBER OF $\bar{x}_{maj}$ POINTS IS 10

(6)

1	2	3	4	5	6	7	8	9
0	2 0							

INTERX(I) DELETE IF NMAJ=1

(7)

ADDITIONAL INPUT FOR KTYPE = 2,3

$\frac{c_m(t_1-t_0)}{b}$	
k	

EXAMPLE 5 INPUT DATA (Cont)

(1)

RUN NO.	KODEN	KTYPE	KODSP	KCO	MAX OUT	KAPP	MSTAP	KAPN	MSTAN	NACRCY	NRO	NTH
3 6	9	12	15	18	21	24	27	30	33	36	39	42
1 6 1 8	0	1	1	- 1	0	1 0 1	4	5 0	6	4	1	3

NMAJ
45
4

INTEGER CARD

(2)

$\frac{h}{R}$	0.1
$\frac{b}{c_{dm}T}$	0.009090909
ν	0.2
ν_m	0.2
$\frac{E_m}{E}$	1.2
$\frac{\gamma}{\gamma_m}$	1.0135135
$\frac{b}{\Omega_2 c_{dm}}$	1.0
$\frac{\Omega_1}{\Omega_2}$	1.0
$\frac{\tau_1}{\Omega_1}$	0.1
$\frac{\tau_2}{\Omega_2}$	0.1
$\frac{c_m t_0}{b}$	33.0

(3)

ρ_1	0.9047619
ρ_2	
ρ_3	

MAXIMUM NUMBER OF ρ, POINTS IS 3

(4)

θ_1	0.0
θ_2	90.0
θ_3	180.0
θ_4	
θ_5	
θ_6	
θ_7	
θ_8	
θ_9	
θ_{10}	

MAXIMUM TOTAL NUMBER OF θ POINTS IS 10

(5)

$\bar{x}_1$	-1.0
$\bar{x}_2$	0.0
$\bar{x}_3$	18.0
$\bar{x}_4$	40.0
$\bar{x}_5$	
$\bar{x}_6$	
$\bar{x}_7$	
$\bar{x}_8$	
$\bar{x}_9$	
$\bar{x}_{10}$	

MAXIMUM NUMBER OF $\bar{X}_{mej}$ POINTS IS 10

(6)

1	2	3	4	5	6	7	8	9
0	0	2 1						

INTERX (I) DELETE IF NMAJ=1

(7)

ADDITIONAL INPUT FOR KTYPE = 2,3

$\frac{c_m(t_1-t_0)}{b}$	
k	

D. Special Considerations for Early Time Response (for Step or Rectangular Pulses)

The accuracy of the early time responses depends on the choice of KAPP, $c_m t_o/b$, and $b/c_{dm}T$. The parameter $c_m t_o/b$ controls the "rest time" while KAPP affects the Fourier representation of the incident wave. The parameter $b/c_{dm}T$ affects the early time response due to a "scaling effect" that can be described as follows: For a finite number of terms the Fourier expansion approximates the step discontinuity in the incident pulse by a steep ramp. The parameter $b/c_{dm}T$ governs the slope of this ramp relative to the size of the cylinder. That is, in the scale of the cylinder, a decrease in the value of this parameter corresponds to a stretching of the ramp region in the pulse. Accordingly the inaccuracies in the cylinder response due to the passage of this ramp over the cylinder will occur over a larger interval of time. By a reverse process, the accuracy of the early time response can be improved, provided that the increase in $b/c_{dm}T$ does not decrease the value of $c_m t_o/b$ and, hence, the rest time, below an acceptable limit. To keep the rest time within this limit may then require getting a refined early time response with a pulse of short duration and then proceeding as outlined previously, in Example 5 to get the complete response. In most practical applications where the maximum response is of greatest importance, the slight inaccuracies introduced at early time are acceptable, and, as in the example problems presented, the above procedure need not be applied.

SECTION III

FORTRAN LISTING OF PROGRAM

```
C MAIN PROGRAM
C     DYNAMIC STRESSES OF THICK CYLINDERS
      DIMENSION KCON1(10,150),D1(10,150),KCON2(3,10,150),D2(3,10,150)
      DIMENSION LCON1(10,30), MCON1(10,30), NCON1(10,30), ICON1(10,30),
     1JCON1(10,30),  LCON2(3,10,30), MCON2(3,10,30), NCON2(3,10,30),
     2ICON2(3,10,30),  A1(10,30), B1(10,30), C1(10,30), G1(10,30),
     3H1(10,30),  A2(3,10,30), B2(3,10,30), C2(3,10,30), G2(3,10,30),
     4H2(3,10,30), JCON2(3,10,30)
      DIMENSION F(3,10),JF(3,10),          COSPT(30),SINPT(30)
       DIMENSION XMAJ(10),KEYTH(10),INTERX(9)
      DIMENSION TAD(30),TBE(30),TCC(30),TGC(30),TH7(30)
      DIMENSION RHO(3),THETA(10),XBAR(30),TITLE(43),E(11)
      DIMENSION REJ(100), EMJ(100), WASTE(100), BJQ1B(100), BJQ2B(100),
     1BJQ1(100), BJQ2(100), BRJ3(100), BIJ3(100), BRJ4(100), BIJ4(100),
     2BJQ1R(100), BJQ2R(100),    BYQ1B(2), BYQ2B(2), BYQ1(2), BYQ2(2),
     3BRY3(2), BIY3(2), BRY4(2), BIY4(2), BYQ1R(2), BYQ2R(2)
      DIMENSION EL(12,12),EM(12,1), X1(100), X2(100), X3(100), X4(100),
     1X5(100), X6(100), X7(100), X8(100), X9(100), X10(100), X11(100),
     2X12(100)
      COMMON KCON1,D1,KCON2,D2
      EQUIVALENCE (KCON1(1),LCON1), (KCON1(301),MCON1), (KCON1(601),
     1NCON1),(KCON1(901),ICON1),  (KCON1(1201),JCON1), (KCON2(1),
     2LCON2),(KCON2(901),MCON2),  (KCON2(1801),NCON2), (KCON2(2701),
     3ICON2),(KCON2(3601),JCON2),  (D1(1),A1), (D1(301),B1), (D1(601),
     4C1),  (D1(901),G1), (D1(1201),H1),   (D2(1),A2), (D2(901),B2),
     5(D2(1801),C2),  (D2(2701),G2),  (D2(3601),H2)
      EQUIVALENCE  (E(1),E1),(E(2),E2),(E(3),E3),(E(4),E4),(E(5),E5),
     1(E(6),E6),(E(7),E7),(E(8),E8),(E(9),E9),(E(10),E10),(E(11),E11)
      EQUIVALENCE (NRO1,NRO),(NTH1,NTH),(NT1,NT)
      EQUIVALFNCE (F,D1),(JF,KCON1)
      EQUIVALENCE (RHO,TITLE),(THETA,TITLE(4)),(XBAR,TITLE(14))
C     INPUT
 2002 FORMAT(A6,20I3)
 1002 FORMAT (19I3)
 1003 FORMAT (E14.7)
 1001 READ INPUT TAPE 7,2002,RUN1,KODEN,KTYPE,KODSP,KCO    ,MAXOUT,KAPP,
     1MSTAP,KAPN,MSTAN,NACRCY,NRO,NTH,NMAJ
      NACY1=NACRCY+1
      TOL = 10.**(-NACY1)
      ILK = IBIG(TOL,0)
      IF (KODEN) 3999,4000,4001
 3999 RUN2 = 6H COMBI
      RUN3 = 6HNED WA
B     RUN4 = 652560606060
      GO TO 4002
 4000 RUN2 = 6H DILAT
      RUN3 = 6HATIONA
      RUN4 = 6HL WAVE
      GO TO 4002
 4001 RUN2 = 6H SHEAR
      RUN3 = 6H WAVE
B     RUN4 = 606060606060
 4002 WRITE OUTPUT TAPE 6,4003,RUN1,RUN2,RUN3,RUN4,KODEN,KTYPE,KODSP,KCO
     1,MAXOUT,KAPP,MSTAP,KAPN,MSTAN,NACRCY,NRO,NTH,NMAJ
 4003 FORMAT (12H1RUN NUMBER A6///A6,A6,A6////6H INPUT//108H        KODE
     1N        KTYPE          KODSP           KCO      MAXOUT        KAPP
     2  MSTAP          KAPN           MSTAN/9I12//30H        NACRCY          NRO
     3          NTH            NMAJ  /4I12//)
      READ INPUT TAPE 7,1003,(E(I),I=1,11), (RHO(I),I=1,NRO), (THETA(I),
     1I=1,NTH), (XMAJ(I),I=1,NMAJ)
      WRITE OUTPUT TAPE 6,1008,(E(I),I=1,11)
```

```
1008 FORMAT (//5X,103H         H / R            B / CDMT                NU
    1                NU M            E M / E          GAM / GAM M/1P6E18.7//5
    2X,85HB OMEGA / COM          OM1 / OM2         TAU1 / OM1          TAU2
    3 / OM2         C M TO / B/1P5E18.7)
     E7 = E7 / E2
     WRITE OUTPUT TAPE 6,1009,(RHO(I),I=1,NRO)
1009 FORMAT (///4H RHO,5X,1P3E18.7)
     WRITE OUTPUT TAPE 6,1010,(THETA(I),I=1,NTH)
1010 FORMAT (//6H THETA,5X,1P5E18.7/11X,1P5E18.7)
     WRITE OUTPUT TAPE 6,1011,(XMAJ(I),I=1,NMAJ)
1011 FORMAT (//11H MAJOR XBAR,5X,1P5E18.7/16X,1P5E18.7)
     NMAJ1 = NMAJ - 1
     IF (NMAJ1) 3001,505,499
 499 READ INPUT TAPE 7,1002,(INTERX(I),I=1,NMAJ1)
     WRITE OUTPUT TAPE 6,1012,(INTERX(I),I = 1,NMAJ1)
1012 FORMAT (//7H INTERX,5X,10I5)
 500 XBAR(1) = XMAJ(1)
     JAY = 1
     DO 504  I =1,NMAJ1
     IF (INTERX(I)) 3001,503,501
 501 INTER = INTERX(I)
     ENX = INTER + 1
     DEL = (XMAJ(I+1) - XMAJ(I)) / ENX
     DO 502  J = 1,INTER
     JAY = JAY + 1
     YAY = J
 502 XBAR(JAY) = XMAJ(I) + YAY*DEL
 503 JAY = JAY + 1
 504 XBAR(JAY) = XMAJ(I+1)
 505 NX = JAY
     FACT2 = E2
     EME=E(5)
     E(5)=EME *((1.0+E(3))/(1.0+E(4)))
     E6=EME*E6*((1.0-E4)/((1.0+E4)*(1.0-2.0*E4)))*((1.0+E3)
    1*(1.0-2.0*E3)/(1.0-E3))
     E(6)=SQRTF(E6)
     E2 = F2 * E6
     E(1)=(2.0-E(1))/(2.0+E(1))
     BETA=E1
     BETASQ=E1*E1
     VMDUM=(1.0-E4)/(1.0-2.0*E4)
     SVMDUM=SQRTF(2.0*VMDUM)
     TR1 = -1.
     TR2 = 0.
     ZETA = BETA
     KEY = -1
     JUMP = 0
     IF (KODEN) 507,511,508
 507 KODEN = 0
     KEY = 1
     REWIND 3
     GO TO 530
 508 IF (KEY) 509,1001,510
 509 KEY = 0
     GO TO 530
 510 KODEN = 1
     GO TO 513
 511 IF (KEY) 512,1001,1001
 512 KEY = 0
     GO TO 530
 513 READ INPUT TAPE 7,1002,LEADX,KEYX,(KEYTH(I),I=1,NTH)
```

```
      READ INPUT TAPE 7,1002,KTYPE,KCO
      READ INPUT TAPE 7,1003,PHI,AMPFAC,E11
      WRITE OUTPUT TAPE 6,4005,LEADX,KEYX,(KEYTH(I),I = 1,NTH)
 4005 FORMAT (35H1ADDITIONAL INPUT FOR COMBINED WAVE//////21H LEADX
     1      KEYX/I6,I15////6H KEYTH/1X,10I5)
      WRITE OUTPUT TAPE 6,4006,KTYPE,KCO
 4006 FORMAT (22H      KTYPE          KCO/I10,I12)
      WRITE OUTPUT TAPE 6,4007,PHI,AMPFAC,E11
 4007 FORMAT (54H              PHI            AMPFAC        C M TO / 8/
     11P3E18.7)
      INTER = LEADX - 1
      IF (INTER) 3001,516,514
  514 DO 515  I = 1,INTER
  515 LEADX = LEADX + INTERX(I)
  516 XSTAR = XMAJ(KEYX)
      NT = NX - LEADX + 1
      INTER = NTH
      J = 0
      DO 518  I = 1,NTH
      IF (KEYTH(I)) 517,518,517
  517 THETA(INTER+1) = -THETA(I)
      J = J + 1
      INTER = INTER + 1
      KEYTH(J) = I
  518 CONTINUE
      NTHOLD = NTH
      NTH = INTER
      NEWTH = NTH - NTHOLD
      DO 519  I = 1,NTH
  519 THETA(I) = THETA(I) - PHI
      GO TO 531
  530 NT = NX
      LEADX = 1
  531 ADJ1 = FACT2
      COUNT = 0.
      TALLY = 0.
      TAP = 0.
      IF (KODEN) 3001,533,532
  532 FACT2 = FACT2 * SVMDUM
      TR1 = 0.
      TR2 = 1.
      ZETA = -BETA
  533 TOT = E11*FACT2
      ADJ2 = TOT
      ANGLE = TOT - 2.*FACT2
      E11 = 1. - 2.*TOT
      IF (KODEN) 3001,537,534
  534 IF (KEY) 3001,535,536
  535 ADJ1 = ADJ1*SVMDUM
      GO TO 537
  536 ADJ2 = ADJ2 - ADJ1*XSTAR
  537 DO 540  I = 1,NTH
  540 THETA(I) = .0174532927 * THETA(I)
      GO TO (8002,550,550),KTYPE
  550 READ INPUT TAPE 7,1003,CTBIN,CAYIN
      EMIN = CAYIN
      T1TOT=CTBIN*FACT2
      T1T=T1TOT+TOT
      T2T = 1. - TOT
      ACAPT=EMIN/(1.0-T1T)
      GO TO (8002,8010,8016),KTYPE
```

```
8002 WRITE OUTPUT TAPE 6,8003
8003 FORMAT (///17H RECTANGULAR WAVE)
     GO TO 8020
8010 WRITE OUTPUT TAPE 6,8011,CTBIN,CAYIN
8011 FORMAT (///16H TRIANGULAR WAVE///18H      C-M(T1-TO)/B,17X,1HK/1P2
    1E18.7)
     GO TO 8020
8016 WRITE OUTPUT TAPE 6,8017,CTBIN,EMIN
8017 FORMAT (///23H EXPONENTIAL DECAY WAVE///6X,12HC-M(T1-TO)/B,17X,1HM
    1/1P2E18.7)
C    CONSTANTS FOR L-MATRIX
8020 PI=3.1415927
     PISQ=9.8696047
     Q1DUM=PI*E2
     Q2DUM=Q1DUM*SQRTF((2.0-2.0*E3)/(1.0-2.0*E3))
     ELOVU=(2.0*E3)/(1.0-2.0*E3)
     CDUM=(0.5-E4)/(1.0-E4)
     Q0BAR=E7**2*E8*(E9-1.3333333*CDUM*E10)
     Q1BAR=E7*(E9*E8+1.0-1.3333333*CDUM*(E10+E8))
     Q2BAR=1.0-1.3333333*CDUM
     VDUM=(1.0-E3)/(1.0-2.0*E3)
     YDUM1=Q0BAR/PISQ
     YDUM2=E7**2/PISQ
     YDUM3=YDUM2*E8
     YDUM4=Q1BAR*E7*(E8+1.0)/PISQ
     YDUM5=YDUM3*E8
     YDUM6=YDUM2*E10
     YDUM7=Q1BAR/PI
     YDUM8=E7*E8/PI
     YDUM9=VDUM/VMDUM
     YDUM10=E7*(1.0-E10)/PI
     Q34DUM=PI*E2*SQRTF(VDUM)/E6
     YDUM11=E5/YDUM9
     DO   9001 IZR=1,NR01
     DO   9001 IZTH=1,NTH1
     DO   9001 IZT=1,NT1
     A2(IZR,IZTH,IZT)=0.0
     LCON2(IZR,IZTH,IZT)=0
     B2(IZR,IZTH,IZT)=0.0
     MCON2(IZR,IZTH,IZT)=0
     G2(IZR,IZTH,IZT)=0.0
     ICON2(IZR,IZTH,IZT)=0
     H2(IZR,IZTH,IZT)=0.0
     JCON2(IZR,IZTH,IZT)=0
     C2(IZR,IZTH,IZT)=0.0
9001 NCON2(IZR,IZTH,IZT)=0
C    MAJOR LOOP ON P
     DO   219  IP=1,KAPP
     GO TO (9002,9003,9003),KTYPE
9002 IF(IP-2*(IP/2))  219,219,9003
9003 P=IP
     PPI=P*PI
     COSST = COSF(PPI*ANGLE)
     SINST = SINF(PPI*ANGLE)
     JX = LEADX
     DO  1     ITT=1,NT
     ANGL = ADJ1*XBAR(JX) + ADJ2
     JX = JX + 1
     SINPT(ITT)=SINF(PPI*ANGL)
   1 COSPT(ITT)=COSF(PPI*ANGL)
C    CALCULATED QUANTITIES
```

```
      Q1=P*Q1DUM
      Q1SQ=Q1*Q1
      Q2=P*Q2DUM
      Q2SQ=Q2*Q2
      HALFSQ = .5 * Q2SQ
      PSQ=P*P
      YVP1=((YDUM1/PSQ)-Q2BAR)*((YDUM3/PSQ)-1.0)+YDUM4/PSQ
      YVP1=YVP1/((YDUM5/PSQ)*((YDUM2/PSQ)+1.0)+(YDUM2/PSQ)+1.0)
      YVP1=2.0*VDUM*YVP1
      YSP1=((YDUM6/PSQ+1.0)/(YDUM2/PSQ+1.0))*YDUM9
      YVP2=(YDUM7/P)*(YDUM3/PSQ-1.0)-(YDUM1/PSQ-Q2BAR)
     1*(YDUM8/P+E7/PPI)
      YVP2=YVP2/((YDUM5/PSQ)*(YDUM2/PSQ+1.0)+YDUM2/PSQ+1.0)
      YVP2=2.0*VDUM*YVP2
      YSP2=((YDUM10/P)/(YDUM2/PSQ+1.0))*YDUM9
      YP2=YVP2+1.33333333*YSP2
      YP1=YVP1+1.33333333*YSP1
      SP=YP2**2/YP1**2
      SP=SQRTF(1.0+SP)
      SSP=YSP2**2/YSP1**2
      SSP=SQRTF(1.0+SSP)
      Q3BAR=SQRTF((1.0+SP)/YP1)
      Q3BAR=Q3BAR*P*Q34DUM/SP
      Q3TIL=((YP2/YP1)/(SP+1.0))*Q3BAR
      Q4BAR=SQRTF((1.0+SSP)/YSP1)
      Q4BAR=Q4BAR*P*Q34DUM/SSP
      Q4TIL=((YSP2/YSP1)/(SSP+1.0))*Q4BAR
      YVP13U=YVP1*YDUM11
      YVP23U=YVP2*YDUM11
      YSP12U=YSP1*YDUM11
      YSP22U=YSP2*YDUM11
      ELBU=YVP13U-0.666666666*YSP12U
      ELTU=YVP23U-0.666666666*YSP22U
      UBU=YSP12U
      UTU=YSP22U
      EPDUM1=ELBU/YDUM11
      EPDUM2=ELTU/YDUM11
      EPSB=EPDUM1* YP1+EPDUM2 *YP2
      EPSB=EPSB/(YP1**2+YP2**2)
      S3=SQRTF(Q3BAR**2+Q3TIL**2)
      S4=SQRTF(Q4BAR**2+Q4TIL**2)
      Q3E=S3
      Q4E=S4
      ANGQ3=Q3TIL/Q3BAR
      ANGQ4=Q4TIL/Q4BAR
      PHI3=-ATANF(ANGQ3)
      PHI4=-ATANF(ANGQ4)
      IF(KODEN)    4031,4031,4032
 4031 ABQ=SQRTF(Q3BAR**2+Q3TIL**2)
      CALL BESSEL (KAPN,ABQ,PHI3,REJ,EMJ,ENG)
      IF(ENG)   4035,4035,3001
 4032 ABQ=SQRTF(Q4BAR**2+Q4TIL**2)
      CALL BESSEL (KAPN,ABQ,PHI4,REJ,EMJ,ENG)
      IF(ENG)   4035,4035,3001
 4035 Q12B=Q1*Q1*E1
      Q1B2=Q12B*E1
      Q22B=Q2*Q2*E1
      Q2B2=Q22B*E1
      Q32D=Q3BAR**2-Q3TIL**2
      Q42D=Q4BAR**2-Q4TIL**2
      BSUM=0.5*ELBU+UBU
```

```
      TSUM=0.5*ELTU+UTU
      Q3P=Q3BAR*Q3TIL
      Q4P=Q4BAR*Q4TIL
      BBSUM=0.5*Q3E*(Q3BAR*UBU+Q3TIL*UTU)
      BTDIF=0.5*Q3E*(Q3BAR*UTU-Q3TIL*UBU)
      EPST=EPDUM2* YP1-EPDUM1 *YP2
      EPST=EPST/(YP1**2+YP2**2)
    5 DO  6     IZOUT=1,KAPN
      BJQ1B(IZOUT)=0.0
      BJQ2B(IZOUT)=0.0
      BJQ1(IZOUT)=0.0
      BJQ2(IZOUT)=0.0
      BRJ3(IZOUT)=0.0
      BIJ3(IZOUT)=0.0
      BRJ4(IZOUT)=0.0
    6 BIJ4(IZOUT)=0.0
C     TILDE FUNCTIONS INDEPENDENT OF ROE
      ENG=0.0
    7 CALL TILDE (KAPN,Q1,BETA,0.0,BJQ1B,WASTE,
     1BYQ1B,WASTE,ENG)
      IF(ENG)  8,8,3001
    8 CALL TILDE (KAPN,Q2,BETA,0.0,BJQ2B,WASTE,
     1BYQ2B,WASTE,ENG)
      H22 = 2. * BYQ2B(2)
      IF(ENG)  9,9,3001
    9 CALL TILDE (KAPN,Q1,1.0,0.0,BJQ1,WASTE,
     1BYQ1,WASTE,ENG)
      IF(ENG)  10,10,3001
   10 CALL TILDE (KAPN,Q2,1.0,0.0,BJQ2,WASTE,
     1BYQ2,WASTE,ENG)
      IF(ENG)  11,11,3001
   11 CALL TILDE (KAPN,S3,1.0,PHI3,BRJ3,BIJ3,
     1BRY3,BIY3,ENG)
      IF(ENG)  12,12,3001
   12 CALL TILDE (KAPN,S4,1.0,PHI4,BRJ4,BIJ4,
     1BRY4,BIY4,ENG)
      IF(ENG)  13,13,3001
   13 NSTAR=0
      KROE=1
C     DO LOOP ON ROE
      DO  210  IROE=1,NROI
C     TILDE FUNCTIONS DEPENDING ON ROE
      ROE = RHO(IROE)
   15 DO  17    IZOUT=1,KAPN
      BJQ1R(IZOUT)=0.0
   17 BJQ2R(IZOUT)=0.0
   18 CALL TILDE (KAPN,Q1,ROE,0.0,BJQ1R,WASTE,
     1BYQ1R    ,WASTE,ENG)
      IF(ENG)  19,19,3001
   19 CALL TILDE (KAPN,Q2,ROE,0.0,BJQ2R,WASTE,
     1BYQ2R    ,WASTE,ENG)
      IF(ENG)  20,20,3001
C     ZERO CASE FOR ALL THETA AND T/T
   20 GO TO(21,9031),KROE
   21 DO  22    IEL=1,5
      DO  22    JEL=1,4
   22 EL(IEL,JEL)=0.0
      IF(KODEN)    9022,9022,7023
 9022 EL(1,1)=Q1SQ*((0.5/E1)*BJQ1B(2)-(0.5*ELOVU+1.0)*BJQ1B(1))
      EL(1,2)=Q1SQ*((2.0/Q12B)*BYQ1B(2)-(0.5*ELOVU+1.0)*BYQ1B(1))
      EL(2,1)=Q1SQ*(0.5*BJQ1(2)-(0.5*ELOVU+1.0)*BJQ1(1))
```

```
      EL(2,2)=Q1SQ*((2.0/Q1SQ)*BYQ1(2)-(0.5*ELOVU+1.0)*BYQ1(1))
      EL231=BSUM*(Q32D*BRJ3(1)+2.0*Q3P*BIJ3(1))
     1-TSUM*(Q32D*BIJ3(1)-2.0*Q3P*BRJ3(1))
      EL232=EL231-(BBSUM*BRJ3(2)-BTDIF*BIJ3(2))
      EL233=BSUM*(Q32D*BIY3(1)-2.0*Q3P*BRY3(1))
     1+TSUM*(Q32D*BRY3(1)+2.0*Q3P*BIY3(1))
      EL234=(2.0/Q3E)*(Q3BAR*(UTU*BRY3(2)+UBU*BIY3(2))+Q3TIL*
     1(UTU*BIY3(2)-UBU*BRY3(2)))
      EL(2,3)=EL232+EL233-EL234
      EL(3,1)=(-Q1SQ)*0.5*BJQ1(2)
      EL(3,2)=(-2.0)*BYQ1(2)
      EL331=0.5*Q3E*(Q3BAR*BRJ3(2)+Q3TIL*BIJ3(2))
      EL332=(2.0/Q3E)*(Q3BAR*BIY3(2)-Q3TIL*BRY3(2))
      EL(3,3)=EL331+EL332
      EL261=BSUM*(Q32D*BIJ3(1)-2.0*Q3P*BRJ3(1))
     1+TSUM*(Q32D*BRJ3(1)+2.0*Q3P*BIJ3(1))
      EL261=EL261-(BTDIF*BRJ3(2)+BBSUM*BIJ3(2))
      EL262=BSUM*(Q32D*BRY3(1)+2.0*Q3P*BIY3(1))
     1+TSUM*(2.0*Q3P*BRY3(1)-Q32D*BIY3(1))
      EL262=-EL262
      EL263=(2.0/Q3E)*(Q3BAR*(UBU*BRY3(2)-UTU*BIY3(2))+
     1Q3TIL*(UBU*BIY3(2)+UTU*BRY3(2)))
      EL(2,6)=EL261+EL262+EL263
      EL(2,6)=-EL(2,6)
      EL361=(-0.5)*Q3E*(Q3BAR*BIJ3(2)-Q3TIL*BRJ3(2))
      EL362=(2.0/Q3E)*(Q3BAR*BRY3(2)+Q3TIL*BIY3(2))
      EL(3,6)=EL361+EL362
      EL(5,3)=-EL(2,6)
      EL(6,3)=-EL(3,6)
      DO 23   IEL=1,3
      IIEL=IEL+3
      DO 23   JEL=1,3
      JJEL=JEL+3
   23 EL(IIEL,JJEL)=EL(IEL,JEL)
      CALL ZNHSP (P,E2,E5,E6,E11,Q3TIL,Q3BAR,EPSB,EPST,VMDUM
     1,KTYPE,KAPP,KODSP,CAYIN,TITOT,TOT,T1T,T2T,REJ,EMJ,EM,ACAPT)
      GO TO 9024
 7023 EL(1,1)=(0.5*Q2SQ)*(BJQ2B(1)-(1.0/E1)*BJQ2B(2))
      EL(1,2)=0.5*Q2SQ*BYQ2B(1)-(2.0/E1)*BYQ2B(2)
      EL(2,1)=(0.5*Q2SQ)*(BJQ2(1)-BJQ2(2))
      EL(2,2)=(0.5*Q2SQ)*(BYQ2(1)-(4.0/Q2SQ)*BYQ2(2))
      EL231=Q4E*(Q4BAR*BRJ4(2)+Q4TIL*BIJ4(2))-Q42D*BRJ4(1)-2.0*Q4P
     1*BIJ4(1)-Q42D*BIY4(1)+2.0*Q4P*BRY4(1)+(4.0/Q4E)
     2*(Q4BAR*BIY4(2)-Q4TIL*BRY4(2))
      EL231=0.5*EL231
      EL232=Q4E*(Q4BAR*BIJ4(2)-Q4TIL*BRJ4(2))-Q42D*BIJ4(1)+2.0*Q4P
     1*BRJ4(1)+Q42D*BRY4(1)+2.0*Q4P*BIY4(1)-(4.0/Q4E)
     2*(Q4BAR*BRY4(2)+Q4TIL*BIY4(2))
      EL232=0.5*EL232
      EL(2,3)=UBU*EL231-UTU*EL232
      EL(3,1) = HALFSQ*BJQ2(2)
      EL(3,2) = 2.*BYQ2(2)
      EL(3,3)=(2.0/Q4E)*(Q4TIL*BRY4(2)-Q4BAR*BIY4(2))
     1-(0.5*Q4E)*(Q4BAR*BRJ4(2)+Q4TIL*BIJ4(2))
      EL(2,6)=(-1.0)*(UTU*EL231+UBU*EL232)
      EL361=(2.0/Q4E)*(Q4BAR*BRY4(2)+Q4TIL*BIY4(2))
     1-(0.5*Q4E)*(Q4BAR*BIJ4(2)-Q4TIL*BRJ4(2))
      EL(3,6)=-EL361
      EL(5,3)=-EL(2,6)
      EL(6,3)=-EL(3,6)
      DO 7024    IEL=1,3
```

```
      IIEL=IEL+3
      DO  7024   JEL=1,3
      JJEL=JEL+3
 7024 EL(IIEL,JJEL)=EL(IEL,JEL)
      CALL ZRHSM (P,E2,E5,E6,E11,Q4TIL,Q4BAR,EPSB,EPST,VMDUM
     1,KTYPE,KAPP,KODSP,CAYIN,T1TOT,TOT,T1T,T2T,REJ,EMJ,EM,ACAPT)
 9024 IF(KCO)      27,27,25
   25 WRITE OUTPUT TAPE 6,26,((EL(I,J),J=1,6 ),I=1,6 ),(EM(K,1),K=1,6)
   26 FORMAT (36H1THE MATRICES FOR N=0 ARE AS FOLLOWS ////
     113H THE L MATRIX //6E17.7/ 6E17.7/ 6E17.7/ 6E17.7/ 6E17.7/
     26E17.7///13H THE M VECTOR // 6E17.7///)
   27 CALL LINSYS (EL,EM,6,LERUR)
      IF(LERUR)     3006,3006,9027
 9027 IF(KCO)       30,30,28
   28 WRITE OUTPUT TAPE 6,29,(EM(K,1),K=1,6)
   29 FORMAT (13H THE X VECTOR // 6E17.7///////)
   30 IF(KODEN)     7031,7031,7030
 7030 X1(1)=0.0
      X2(1) = 0.
      X3(1) = EM(1,1)
      X4(1)=EM(2,1)
      X5(1)=0.0
      X6(1)=EM(3,1)
      X7(1)=0.0
      X8(1) = 0.
      X9(1) = EM(4,1)
      X10(1)=EM(5,1)
      X11(1)=0.0
      X12(1)=EM(6,1)
      GO TO 7032
 7031 X1(1)=EM(1,1)
      X2(1)=EM(2,1)
      X3(1)=0.0
      X4(1)=0.0
      X5(1)=EM(3,1)
      X6(1)=0.0
      X7(1)=EM(4,1)
      X8(1)=EM(5,1)
      X9(1)=0.0
      X10(1)=0.0
      X11(1)=EM(6,1)
      X12(1)=0.0
 7032 KROE=2
 9031 IF(KCO)  9034,9034,9032
 9032 WRITE OUTPUT TAPE 6,9033,ROE
 9033 FORMAT (5H1ROE=E14.7,5X,39HMATRIX-XBAR,SRR,STT,SRT,UR,UT FOR N=0
     1//)
 9034 A11=Q1SQ*(0.5*ELOVU+1.0)
      A12=1.0/ROE
      A13=A11*(BJQ1R(1)*X1(1)+BYQ1R(1)*X2(1))-Q1SQ*A12*(BJQ1R(2)*X1(1)
     1*0.5+(2.0/Q1SQ)*BYQ1R(2)*X2(1))
      A14=A11*(BJQ1R(1)*X7(1)+BYQ1R(1)*X8(1))-Q1SQ*A12*(BJQ1R(2)*X7(1)
     1*0.5+(2.0/Q1SQ)*BYQ1R(2)*X8(1))
      B11 = .5 * ELOVU * Q1SQ * (BJQ1R(1)*X1(1) + BYQ1R(1)*X2(1))
      B12=A12*((Q1SQ/2.0)*BJQ1R(2)*X1(1)+2.0*BYQ1R(2)*X2(1))
      B13 = .5 * ELOVU * Q1SQ * (BJQ1R(1)*X7(1) + BYQ1R(1)*X8(1))
      B14=A12*((Q1SQ/2.0)*BJQ1R(2)*X7(1)+2.0*BYQ1R(2)*X8(1))
      C11 = BJQ2R(1) - BJQ2R(2)/ROE
      C12 = BYQ2R(1) - 4.*BYQ2R(2)/(Q2SQ*ROE)
      C13 = HALFSQ * (C11*X3(1) + C12*X4(1))
      C14 = HALFSQ * (C11*X9(1) + C12*X10(1))
```

```
      G11=(Q1SQ/2.0)*BJQ1R(2)*X1(1)+2.0*BYQ1R(2)*X2(1)
      G12=(Q1SQ/2.0)*BJQ1R(2)*X7(1)+2.0*BYQ1R(2)*X8(1)
      H11 = HALFSQ * BJQ2R(2)
      H12 = 2. * BYQ2R(2)
      H13 = H11 * X3(1) + H12 * X4(1)
      H14 = H11 * X9(1) + H12 * X10(1)
      H21 = HALFSQ * BJQ2B(2)
      H23 = H21 * X3(1) + H22 * X4(1)
      H24 = H21 * X9(1) + H22 * X10(1)
      H25 = H23 * COSST - H24 * SINST
      DO  44   ITH=1,NTH1
   33 IF(KCO)  36,36,34
   34 THET = 57.2957796*THETA(ITH)
      WRITE OUTPUT TAPE 6,35,THET
   35 FORMAT (////  7H THETA=E14.7 //)
   36 KSTART=1
      KSTOP=6
      KWASTE=1
      IND1 = ITH
      IND2 = 3*ITH - 3 + IROE
      DO  44   ITT=1,NT1
      IF(NT1-ITT)    37,37,38
   37 KSTOP=NT1
   38 WASTE(KWASTE) = XBAR(ITT)
      KWASTE=KWASTE+1
      DO 410  INK = 1,5
      KON = KCON2(IND2)
      IF (KON) 401,402,402
  401 D1(IND1) = 0.
      GO TO 409
  402 GO TO (403,404,405,406,407),INK
  403 A15=A13*COSPT(ITT)-A14*SINPT(ITT)
      D1(IND1) = A15 * TR1
      GO TO 408
  404 B15=(B11+B12)*COSPT(ITT)-(B13+B14)*SINPT(ITT)
      D1(IND1) = B15 * TR1
      GO TO 408
  405 C15 = C13*COSPT(ITT) - C14*SINPT(ITT)
      D1(IND1) = C15 * TR2
      GO TO 408
  406 G15=G11*COSPT(ITT)-G12*SINPT(ITT)
      D1(IND1) = G15 * TR1
      GO TO 408
  407 H15 = H13 * COSPT(ITT) - H14 * SINPT(ITT) - H25
      D1(IND1) = H15 * TR2
  408 KCON2(IND2) = IBIG(D1(IND1),KON)
  409 IND1 = IND1 + 300
  410 IND2 = IND2 + 900
      IND1 = IND1 - 1490
      IND2 = IND2 - 4470
      IF(KWASTE-7)      9039,40,40
 9039 IF(KSTOP-ITT)   44,40,44
   40 KWASTE=1
      IF(KCO)     44,44,41
   41 WRITE OUTPUT TAPE 6,42,(WASTE(I1),I1=1,6)
   42 FORMAT (5X,6E17.7)
      WRITE OUTPUT TAPE 6,42,(A1(     ITH,I2),I2=KSTART,KSTOP)
      WRITE OUTPUT TAPE 6,42,(B1(     ITH,I3),I3=KSTART,KSTOP)
      WRITE OUTPUT TAPE 6,42,(C1(     ITH,I4),I4=KSTART,KSTOP)
      WRITE OUTPUT TAPE 6,42,(G1(     ITH,I5),I5=KSTART,KSTOP)
      WRITE OUTPUT TAPE 6,42,(H1(     ITH,I6),I6=KSTART,KSTOP)
```

```
      KSTART=KSTART+6
      KSTOP=KSTART+5
      WRITE OUTPUT TAPE 6,43
   43 FORMAT (///)
   44 CONTINUE
      DO  45    IZER=1,NTH1
      DO  45    JZER=1,NT1
      LCON1(       IZER,JZER)=0
      MCON1(       IZER,JZER)=0
      NCON1(       IZER,JZER)=0
      ICON1(       IZER,JZER)=0
   45 JCON1(       IZER,JZER)=0
      NSTOP=KAPN
      F35=1.0
      F36=1.0
C     LOOP ON N
      DO  140  N=1,NSTOP
      IF(KODEN)  9047,9047,9046
 9046 FN=-N
      GO TO 9048
 9047 FN=N
 9048 EN=N
      N1=N+1
      N2=N+2
      EN1=EN+1.0
      ENM1=EN-1.0
      IF(N-NSTAR)       59,59,46
   46 BYD1=BYQ1B(2)
      BYQ1B(2)=(EN/(EN1*BETA))*BYQ1B(2)-((0.25*Q1SQ)/(EN*EN1))*BYQ1B(1)
      BYQ1B(1)=BYD1
      BYD2=BYQ2B(2)
      BYQ2B(2)=(EN/(EN1*BETA))*BYQ2B(2)-((0.25*Q2SQ)/(EN*EN1))*BYQ2B(1)
      BYQ2B(1)=BYD2
 9050 BYD3=BYQ1(2)
      BYQ1(2)=(EN/EN1)*BYQ1(2)-(0.25/(EN*EN1))*BYQ1(1)*Q1SQ
      BYQ1(1)=BYD3
      BYD4=BYQ2(2)
      BYQ2(2)=(EN/EN1)*BYQ2(2)-(0.25/(EN*EN1))*BYQ2(1)*Q2SQ
      BYQ2(1)=BYD4
      BYD5=BRY3(2)
      BRY3(2)=(Q3E/S3)*(EN/EN1)*(COSF(PHI3)*BRY3(2)+SINF(PHI3)*BIY3(2))
     1-((0.25*Q3E**2)/(EN*EN1))*BRY3(1)
      BRY3(1)=BYD5
      BYD6=BIY3(2)
      BIY3(2)=(Q3E/S3)*(EN/EN1)*(COSF(PHI3)*BIY3(2)-SINF(PHI3)*BRY3(1))
     1-((0.25*Q3E**2)/(EN*EN1))*BIY3(1)
      BIY3(1)=BYD6
      BYD7=BRY4(2)
      BRY4(2)=(Q4E/S4)*(EN/EN1)*(COSF(PHI4)*BRY4(2)+SINF(PHI4)*BIY4(2))
     1-((0.25*Q4E**2)/(EN*EN1))*BRY4(1)
      BRY4(1)=BYD7
      BYD8=BIY4(2)
      BIY4(2)=(Q4E/S4)*(EN/EN1)*(COSF(PHI4)*BIY4(2)-SINF(PHI4)*BRY4(1))
     1-((0.25*Q4E**2)/(EN*EN1))*BIY4(1)
      BIY4(1)=BYD8
C     COMPUTE THE L-MATRIX
      ELM1=0.5*ELOVU+1.0-EN*ENM1/Q1B2
      ELM2=(2.0*EN*ENM1/Q2B2)-1.0
      ELM3=0.5*ELOVU+1.0-EN*ENM1/Q1SQ
      ELM4=(2.0*EN*ENM1/Q2SQ)-1.0
      F35=F35*(Q3E/(2.0*EN))**2
```

```
      F36=F36*(Q4E/(2.0*FN))**2
      EL(1,1)=ELM1*BJQ1B(N1)-BJQ1B( N2 )/(2.0*E1*EN1)
      EL(1,1)=(-Q1SQ)*EL(1,1)
      EL(1,2)=ELM1*BYQ1B( 1)-BYQ1B(  2 )*2.0*EN1/Q12B
      EL(1,2)=(-Q1SQ)*EL(1,2)
      EL(1,3)=ENM1*BJQ2B(N1)-Q22B*BJQ2B(N2)/(2.0*EN1)
      EL(1,3)=(FN/BETASQ)*EL(1,3)
      EL(1,4)=ENM1*BYQ2B( 1)-2.0*E1*EN1*BYQ2B( 2)
      EL(1,4)=(FN/BETASQ)*EL(1,4)
      DO  47       KM=1,2
      DO  47       LM=5,12
   47 EL(KM,LM)=0.0
      EL(2,1)=Q12B*BJQ1B( N2 )/(2.0*EN1)-ENM1*BJQ1B(N1)
      EL(2,1)=(FN/BETASQ)*EL(2,1)
      EL(2,2)=2.0*E1*EN1*BYQ1B(  2 )-ENM1*BYQ1B( 1)
      EL(2,2)=(FN/BETASQ)*EL(2,2)
      EL(2,3)=ELM2*BJQ2B(N1)+BJQ2B( N2 )/(E1*EN1)
      EL(2,3)=(-0.5)*Q2SQ*EL(2,3)
      EL(2,4)=ELM2*BYQ2B( 1)+4.0*EN1*BYQ2B(  2 )/Q22B
      EL(2,4)=(-0.5)*Q2SQ*EL(2,4)
      EL(3,1)=ELM3*BJQ1(N1)-BJQ1( N2 )*0.5/EN1
      EL(3,1)=-Q1SQ*EL(3,1)
      EL(3,2)=ELM3*BYQ1( 1)-BYQ1(  2 )*2.0*EN1/Q1SQ
      EL(3,2)=-Q1SQ*EL(3,2)
      EL(3,3)=ENM1*BJQ2(N1)-Q2SQ*BJQ2( N2 )*0.5/EN1
      EL(3,3)=FN*EL(3,3)
      EL(3,4)=ENM1*BYQ2( 1)-2.0*EN1*BYQ2( 2)
      EL(3,4)=FN*EL(3,4)
      EL35=BSUM*(Q32D*BRJ3(N1)+2.0*Q3P*BIJ3(N1))
     1-TSUM*(Q32D*BIJ3(N1)-2.0*Q3P*BRJ3(N1))
      EL351=EL35-(BBSUM*BRJ3( N2 )-BTDIF*BIJ3( N2 ))/EN1
     1-EN*ENM1*(UBU*BRJ3(N1)-UTU*BIJ3(N1))
      EL351=F35*EL351
      EL352=BSUM*(Q32D*BIY3( 1)-2.0*Q3P*BRY3( 1))
     1+TSUM*(Q32D*BRY3( 1)+2.0*Q3P*BIY3( 1))
      EL353=EN*ENM1*(UBU*BIY3( 1)+UTU*BRY3( 1))+(2.0/Q3E)*EN1
     1*(Q3BAR*(UTU*BRY3( 2)+UBU*BIY3( 2))+Q3TIL*(UTU*BIY3( 2)
     2-UBU*BRY3( 2)))
      EL(3,5)=EL351+EL352-EL353
      EL361=ENM1*BRJ4(N1)-0.5*(Q4E/EN1)*(Q4BAR*BRJ4(N2)
     1+Q4TIL*BIJ4(N2))
      EL361=F36*EL361
      EL362=ENM1*BIY4( 1)-2.0*(EN1/Q4E)*(Q4BAR*BIY4( 2)
     1-Q4TIL*BRY4( 2))
      RRRS=EL361+EL362
      RRRS=-RRRS
      EL363=ENM1*BIJ4(N1)-0.5*(Q4E/EN1)*(Q4BAR*BIJ4(N2)
     1-Q4TIL*BRJ4(N2))
      EL363=F36*EL363
      EL364=ENM1*BRY4( 1)-2.0*(EN1/Q4E)*(Q4BAR*BRY4( 2)
     1+Q4TIL*BIY4( 2))
      EIRRS=EL364-EL363
      EL(3,6)=UBU*RRRS-UTU*EIRRS
      EL(3,6)=FN*EL(3,6)
      DO  48    KM=3,6
      DO  48    LM=7,10
   48 EL(KM,LM)=0.0
      EL95=BSUM*(Q32D*BIJ3(N1)-2.0*Q3P*BRJ3(N1))
     1+TSUM*(Q32D*BRJ3(N1)+2.0*Q3P*BIJ3(N1))
      EL951=EL95-(BTDIF*BRJ3( N2 )+BBSUM*BIJ3( N2 ))/EN1
     1-EN*ENM1*(UBU*BIJ3(N1)+UTU*BRJ3(N1))
```

```
      EL951=F35*EL951
      EL952=BSUM*(Q32D*BRY3( 1)+2.0*Q3P*BIY3( 1))
     1+TSUM*(2.0*Q3P*BRY3( 1)-Q32D*BIY3( 1))
      EL952=-EL952
      EL953=EN*ENM1*(UBU*BRY3( 1)-UTU*BIY3( 1))+(2.0/Q3E)*EN1
     1*(Q3BAR*(UBU*BRY3( 2)-UTU*BIY3( 2))+Q3TIL*(UBU*BIY3( 2)
     2+UTU*BRY3( 2)))
      EL(9,5)=EL951+EL952+EL953
      EL(3,11)=-EL(9,5)
      EL(9,6)=UTU*RRRS+UBU*EIRRS
      EL(9,6)=FN*EL(9,6)
      EL(3,12)=-EL(9,6)
      EL(4,1)=(Q1SQ*BJQ1(N2))/(2.0*EN1)-ENM1*BJQ1(N1)
      EL(4,1)=FN*EL(4,1)
      EL(4,2)=2.0*EN1*BYQ1( 2)-ENM1*BYQ1( 1)
      EL(4,2)=FN*EL(4,2)
      EL(4,3)=-(Q2SQ/2.0)*(ELM4*BJQ2(N1)+BJQ2(N2)/EN1)
      EL(4,4)=-(Q2SQ/2.0)*(ELM4*BYQ2( 1)+(4.0/Q2SQ)*EN1*BYQ2( 2))
      EL451=ENM1*BRJ3(N1)-0.5*(Q3E/EN1)*(Q3BAR*BRJ3(N2)
     1+Q3TIL*BIJ3(N2))
      EL451=F35*EL451
      EL452=ENM1*BIY3( 1)-2.0*(EN1/Q3E)*(Q3BAR*BIY3( 2)
     1-Q3TIL*BRY3( 2))
      RRTD=EL451+EL452
      EL453=ENM1*BIJ3(N1)-0.5*(Q3E/EN1)*(Q3BAR*BIJ3(N2)
     1-Q3TIL*BRJ3(N2))
      EL453=F35*EL453
      EL454=ENM1*BRY3( 1)-2.0*(EN1/Q3E)*(Q3BAR*BRY3( 2)
     1+Q3TIL*BIY3( 2))
      EIRTD=EL453-EL454
      EL(4,5)=UBU*RRTD-UTU*EIRTD
      EL(4,5)=FN*EL(4,5)
      EL461=(2.0*EN*ENM1-Q42D)*BRJ4(N1)-2.0*Q4P*BIJ4(N1)
     1+(Q4E/EN1)*(Q4BAR*BRJ4(N2)+Q4TIL*BIJ4(N2))
      EL461=F36*EL461
      EL462=(2.0*EN*ENM1-Q42D)*BIY4( 1)+2.0*Q4P*BRY4( 1)
     1+(4.0*EN1/Q4E)*(Q4BAR*BIY4( 2)-Q4TIL*BRY4( 2))
      RRTS=0.5*(EL461+EL462)
      EL463=(2.0*EN*ENM1-Q42D)*BIJ4(N1)+2.0*Q4P*BRJ4(N1)
     1+(Q4E/EN1)*(Q4BAR*BIJ4(N2)-Q4TIL*BRJ4(N2))
      EL463=F36*EL463
      EL464=(Q42D-2.0*EN*ENM1)*BRY4( 1)+2.0*Q4P*BIY4( 1)
     1-(4.0*EN1/Q4E)*(Q4BAR*BRY4( 2)+Q4TIL*BIY4( 2))
      EIRTS=0.5*(EL463+EL464)
      EL(4,6)=UBU*RRTS-UTU*EIRTS
      EL(10,5)=FN*(UTU*RRTD+UBU*EIRTD)
      EL(4,11)=-EL(10,5)
      EL(10,6)=UTU*RRTS+UBU*EIRTS
      EL(4,12)=-EL(10,6)
      EL(5,1)=EN*BJQ1(N1)-(Q1SQ/EN1)*0.5*BJQ1(N2)
      EL(5,2)=EN*BYQ1( 1)-2.0*EN1*BYQ1( 2)
      EL(5,3)=FN*BJQ2(N1)
      EL(5,4)=FN*BYQ2( 1)
      EL551=EN*BRJ3(N1)-0.5*(Q3E/EN1)*(Q3BAR*BRJ3(N2)
     1+Q3TIL*BIJ3(N2))
      EL551=F35*EL551
      EL552=EN*BIY3( 1)-(2.0*EN1/Q3E)*(Q3BAR*BIY3( 2)
     1-Q3TIL*BRY3( 2))
      EL(5,5)=(-1.0)*(EL551+EL552)
      EL(5,6)=(-FN)*(F36*BRJ4(N1)+BIY4( 1))
      EL5111=EN*BIJ3(N1)-0.5*(Q3E/EN1)*(Q3BAR*BIJ3(N2)
```

```
     1-Q3TIL*BRJ3(N2))
      EL5111=F35*EL5111
      EL5112=(2.0*EN1/Q3E)*(Q3BAR*BRY3( 2)+Q3TIL*BIY3( 2))
     1-EN*BRY3( 1)
      EL(5,11)=EL5111+EL5112
      EL(5,12)=FN*(F36*BIJ4(N1)-BRY4( 1))
      EL(6,1)=(-FN)*BJQ1(N1)
      EL(6,2)=(-FN)*BYQ1( 1)
      EL(6,3)=(-EN)*BJQ2(N1)+0.5*(Q2SQ/EN1)*BJQ2(N2)
      EL(6,4)=(-EN)*BYQ2( 1)+2.0*EN1*BYQ2( 2)
      EL(6,5)=FN*(F35*BRJ3(N1)+BIY3( 1))
      EL661=EN*BRJ4(N1)-0.5*(Q4E/EN1)*(Q4BAR*BRJ4(N2)
     1+Q4TIL*BIJ4(N2))
      EL661=F36*EL661
      EL662=EN*BIY4( 1)-2.0*(EN1/Q4E)*(Q4BAR*BIY4( 2)
     1-Q4TIL*BRY4( 2))
      EL(6,6)=EL661+EL662
      EL(6,11)=FN*(BRY3( 1)-F35*BIJ3(N1))
      EL6121=EN*BIJ4(N1)-0.5*(Q4E/EN1)*(Q4BAR*BIJ4(N2)
     1-Q4TIL*BRJ4(N2))
      EL6121=(-F36)*EL6121
      EL6122=FN*BRY4( 1)-2.0*(EN1/Q4E)*(Q4BAR*BRY4( 2)
     1+Q4TIL*BIY4( 2))
      EL(6,12)=EL6121+EL6122
      DO  51   KM=7,12
      KM6=KM-6
      DO  49   LM=1,6
      LM61=LM+6
   49 EL(KM,LM)=-EL(KM6,LM61)
      DO  50   LM=7,12
      LM62=LM-6
   50 EL(KM,LM)=EL(KM6,LM62)
   51 CONTINUE
      NSTAR=N
      IF(KODEN)     9051,9051,9052
 9051 CALL RHSP  (P,E2,E5,E6,E11,Q3TIL,Q3BAR,EPSR,EPST,VMDUM
     1,KTYPE,KAPP,KODSP,CAYIN,T1TOT,TOT,T1T,T2T,REJ,EMJ,ACAPT,N,EM)
      GO TO 9053
 9052 CALL RHSM  (P,E2,E5,E6,E11,Q4TIL,Q4BAR,EPSB,EPST,VMDUM
     1,KTYPE,KAPP,KODSP,CAYIN,T1TOT,TOT,T1T,T2T,REJ,EMJ,ACAPT,N,EM)
 9053 IF(KCO)       55,55,53
   53 WRITE OUTPUT TAPE 6,54,N,((EL(I,J),J=1,12),I=1,12),(EM(K,1)
     1,K=1,12)
   54 FORMAT (20H1THE MATRICES FOR N=I4,6H   ARE /////
     113H THE L MATRIX //6E17.7/ 6E17.7//6E17.7/ 6E17.7//6E17.7/
     26E17.7//6E17.7/ 6E17.7//6E17.7/ 6E17.7//6E17.7/ 6E17.7//6E17.7/
     36E17.7//6E17.7/ 6E17.7//6E17.7/ 6E17.7//6E17.7/ 6E17.7//6E17.7/
     46E17.7//6E17.7/ 6E17.7/////13H THE M VECTOR // 6E17.7/ 6E17.7/////)
   55 CALL LINSYS (EL,EM,12,LEROR)
      IF(LEROR)        3006,3006,9055
 9055 IF(KCO)       58,58,56
   56 WRITE OUTPUT TAPE 6,57,(EM(K,1),K=1,12)
   57 FORMAT (13H THE X VECTOR // 6E17.7/6E17.7 ///////)
   58 DO  9058 IMT=1,12
      IF(EM(IMT,1))    9059,9058,9059
 9058 CONTINUE
      NSTOP=N
      GO TO 160
 9059 X1(N1)=EM(1,1)
      X2(N1)=EM(2,1)
      X3(N1)=EM(3,1)
```

```
      X4(N1)=EM(4,1)
      X5(N1)=EM(5,1)
      X6(N1)=EM(6,1)
      X7(N1)=EM(7,1)
      X8(N1)=EM(8,1)
      X9(N1)=EM(9,1)
      X10(N1)=EM(10,1)
      X11(N1)=EM(11,1)
      X12(N1)=EM(12,1)
      IF (N - 1) 9049,9049,59
 9049 YQ1B1 = BYQ1B(1)
      YQ1B2 = BYQ1B(2)
      YQ2B1 = BYQ2B(1)
      YQ2B2 = BYQ2B(2)
      T1 = BJQ1B(2)/BETA - Q1SQ*BJQ1B(3)/4.
      T2 = YQ1B1/BETA - 4.*YQ1B2
      T3 = BJQ2B(2)/ZETA
      T4 = YQ2B1/ZETA
      TA = T1*X1(2) + T2*X2(2) + T3*X3(2) + T4*X4(2)
      TB = T1*X7(2) + T2*X8(2) + T3*X9(2) + T4*X10(2)
      TG = TA*COSST - TB*SINST
      T1 = BJQ1B(2)/ZETA
      T2 = YQ1B1/ZETA
      T3 = BJQ2B(2)/BETA - Q2SQ*BJQ2B(3)/4.
      T4 = YQ2B1/BETA - 4.*YQ2B2
      TA = T1*X1(2) + T2*X2(2) + T3*X3(2) + T4*X4(2)
      TB = T1*X7(2) + T2*X8(2) + T3*X9(2) + T4*X10(2)
      TH = TA*COSST - TB*SINST
   59 IF(KCO)    62,62,60
   60 WRITE OUTPUT TAPE 6,61,ROE,N
   61 FORMAT (5H1ROE=E14.7,5X,37HMATRIX-XBAR,SRR,STT,SRT,UR,UY FOR N=
     1I4//)
   62 BYD9=BYQ1R(2)
      BYQ1R(2)=(EN/(EN1*ROE))*BYQ1R(2)-((0.25*Q1SQ)/(EN*EN1))*BYQ1R(1)
      BYQ1R(1)=BYD9
      BYD10=BYQ2R(2)
      BYQ2R(2)=(EN/(EN1*ROE))*BYQ2R(2)-((0.25*Q2SQ)/(EN*EN1))*BYQ2R(1)
      BYQ2R(1)=BYD10
      TAA=0.5*ELOVU-(EN*ENM1)/(Q1SQ*ROE*ROE)+1.0
      TA1=Q1SQ*(TAA*BJQ1R(N1)-(0.5/(EN1*ROE))*BJQ1R(N2))
      TA2=Q1SQ*(TAA*BYQ1R( 1)-((2.0*EN1)/(Q1SQ*ROE))*BYQ1R( 2))
      TA3=(EN/ROE**2)*(ENM1*BJQ2R(N1)-((Q2SQ*ROE)/(2.0*EN1))*BJQ2R(N2))
      TA4=(EN/ROE**2)*(ENM1*BYQ2R( 1)-2.0*EN1*ROE*BYQ2R( 2))
      TAB=TA1*X1(N1)+TA2*X2(N1)-TA3*X3(N1)-TA4*X4(N1)
      TAC=TA1*X7(N1)+TA2*X8(N1)-TA3*X9(N1)-TA4*X10(N1)
      TBB=0.5*ELOVU*Q1SQ+EN*ENM1/ROE**2
      TB1=TBB*BJQ1R(N1)+(Q1SQ/(2.0*EN1*ROE))*BJQ1R(N2)
      TB2=TBB*BYQ1R( 1)+(2.0*EN1/ROE)*BYQ1R( 2)
      TB3=(EN/ROE**2)*(ENM1*BJQ2R(N1)-((Q2SQ*ROE)/(2.0*EN1))*BJQ2R(N2))
      TB4=(EN/ROE**2)*(ENM1*BYQ2R( 1)-2.0*EN1*ROE*BYQ2R( 2))
      TBC=TB1*X1(N1)+TB2*X2(N1)+TB3*X3(N1)+TB4*X4(N1)
      TBD=TB1*X7(N1)+TB2*X8(N1)+TB3*X9(N1)+TB4*X10(N1)
      Q1SR = Q1SQ * ROE
      TC1=(EN/ROE**2)*((Q1SR/(2.0*EN1))*BJQ1R(N2)-ENM1*BJQ1R(N1))
      TC2=(EN/ROE**2)*(2.0*EN1*ROE*BYQ1R( 2)-ENM1*BYQ1R( 1))
      TC3=(0.5*Q2SQ)*((2.0*EN*ENM1/(Q2SQ*ROE**2))*BJQ2R(N1)-BJQ2R(N1)
     1+(1.0/(EN1*ROE))*BJQ2R(N2))
      TC4=(0.5*Q2SQ)*((2.0*EN*ENM1/(Q2SQ*ROE**2))*BYQ2R( 1)-BYQ2R( 1)
     1+((4.0*EN1)/(Q2SQ*ROE))*BYQ2R( 2))
      TCA=TC1*X1(N1)+TC2*X2(N1)-TC3*X3(N1)-TC4*X4(N1)
      TCB=TC1*X7(N1)+TC2*X8(N1)-TC3*X9(N1)-TC4*X10(N1)
```

```
      TG1=(EN/ROE)*BJQ1R(N1)-(Q1SQ/(2.0*EN1))*BJQ1R(N2)
      TG2=(EN/ROE)*BYQ1R(1)-2.0*EN1*BYQ1R(2)
      TG3=(EN/ROE)*BJQ2R(N1)
      TG4=(EN/ROE)*BYQ2R(1)
      TGA=TG1*X1(N1)+TG2*X2(N1)+TG3*X3(N1)+TG4*X4(N1)
      TGB=TG1*X7(N1)+TG2*X8(N1)+TG3*X9(N1)+TG4*X10(N1)
      TH1=(EN/ROE)*BJQ1R(N1)
      TH2=(EN/ROE)*BYQ1R(1)
      TH3=(EN/ROE)*BJQ2R(N1)-((Q2SQ    )/(2.0*EN1))*BJQ2R(N2)
      TH4=(EN/ROE)*BYQ2R(1)-2.0*EN1    *BYQ2R(2)
      TH5=TH1*X1(N1)+TH2*X2(N1)+TH3*X3(N1)+TH4*X4(N1)
      TH6=TH1*X7(N1)+TH2*X8(N1)+TH3*X9(N1)+TH4*X10(N1)
      DO 9107  ITT = 1,NT1
      TAD(ITT) = TAC*SINPT(ITT) - TAB*COSPT(ITT)
      TBE(ITT) = TBD*SINPT(ITT) - TBC*COSPT(ITT)
      TCC(ITT) = TCA*COSPT(ITT) - TCB*SINPT(ITT)
      TGC(ITT) = TGA*COSPT(ITT) - TGB*SINPT(ITT)
      IF (N - 1) 9104,9104,9105
 9104 TGC(ITT) = TGC(ITT) - TG
 9105 TH7(ITT) = TH5*COSPT(ITT) - TH6*SINPT(ITT)
      IF (N - 1) 9106,9106,9107
 9106 TH7(ITT) = TH7(ITT) - TH
 9107 CONTINUE
      DO    129  ITH=1,NTH1
      THET = THETA(ITH)
      THAT = 57.2957796*THET
   65 IF(KCO)     68,68,66
   66 WRITE OUTPUT TAPE 6,67,THAT
   67 FORMAT (////  7H THETA=E14.7 //)
   68 KSTART=1
      KSTOP=6
      KWASTE=1
      IF(KODEN)    9074,9074,9075
 9074 ENDUM=N
      TRIG1=COSF(ENDUM*THET )
      TRIG2=SINF(ENDUM*THET )
      GO TO 9076
 9075 ENDUM=N
      TRIG1=SINF(ENDUM*THET )
      TRIG2=COSF(ENDUM*THET )
 9076 IND1 = ITH
      IND2 = 3*ITH - 3 + IROE
      DO  129        ITT=1,NT1
C     CONVERGENCE TESTS
      DO 110  INK = 1,5
      KON2 = KCON2(IND2)
      IF (KON2) 9077,73,73
   73 KON1 = KCON1(IND1)
      IF (KON1) 109,74,74
   74 GO TO (75,76,77,78,79),INK
   75 TERM = TAD(ITT)*TRIG1
      GO TO 99
   76 TERM = TBE(ITT)*TRIG1
      GO TO 99
   77 TERM = TCC(ITT)*TRIG2
      GO TO 99
   78 TERM = TGC(ITT)*TRIG1
      GO TO 99
   79 TERM =-TH7(ITT)*TRIG2
   99 COUNT = COUNT + 1.
      SUM = D1(IND1) + TERM
```

```
      D1(IND1) = SUM
      KCON2(IND2) = IBIG(SUM,KON2)
      IF (ABSF(TERM/SUM) - TOL) 101,101,100
  100 IF (ABSF(TERM)/TWOK(KON2) - .5E-07) 101,101,105
  101 KON1 = KON1 + 1
      IF (MSTAN - KON1) 102,102,103
  102 KUN1 = -N
      GO TO 108
  103 IF (NSTOP - N) 104,104,108
  104 TALLY = TALLY + 1.
      GO TO 108
  105 KON1 = 0
      IF (NSTOP - N) 106,106,108
  106 TAP = TAP + 1.
 9077 KON1 = -1
  108 KCON1(IND1) = KON1
  109 IND1 = IND1 + 300
  110 IND2 = IND2 + 900
      IND1 = IND1 - 1490
      IND2 = IND2 - 4470
      IF (KCO) 129,129,9078
 9078 IF(NT1-ITT)    69,69,71
   69 KSTOP=NT1
   71 WASTE(KWASTE) = XBAR(ITT)
      KWASTE=KWASTE+1
  124 IF(KWASTE-7)         9124,125,125
 9124 IF(KSTOP-ITT)    129,125,129
  125 KWASTE=1
      IF(KCO)              129,129,126
  126 WRITE OUTPUT TAPE 6,127,(WASTE(I1),I1=1,6)
  127 FORMAT (5X,6E17.7)
      WRITE OUTPUT TAPE 6,127,(A1(      ITH,I2),I2=KSTART,KSTOP)
      WRITE OUTPUT TAPE 6,127,(B1(      ITH,I3),I3=KSTART,KSTOP)
      WRITE OUTPUT TAPE 6,127,(C1(      ITH,I4),I4=KSTART,KSTOP)
      WRITE OUTPUT TAPE 6,127,(G1(      ITH,I5),I5=KSTART,KSTOP)
      WRITE OUTPUT TAPE 6,127,(H1(      ITH,I6),I6=KSTART,KSTOP)
      KSTART=KSTART+6
      KSYOP=KSTART+5
      WRITE OUTPUT TAPE 6,128
  128 FORMAT  (///)
  129 CONTINUE
      DO  135   ICK=1,NTH1
      IND1 = ICK
      DO  135   JCK=1,NT1
      DO 134  INK = 1,5
      IF (KCON1(IND1)) 134,140,140
  134 IND1 = IND1 + 300
  135 IND1 = IND1 - 1490
      GO TO 160
  140 CONTINUE
C     PARTIAL SUMS ON P
  160 DO  210      ITH=1,NTH1
      DO  210      ITT=1,NT1
      INC = ITT
      DO 209  INK = 1,5
      KON = KCON2(IROE,ITH,INC)
      IF (KON) 209,200,200
  200 TERM = D1(ITH,INC)
      SUM = D2(IROE,ITH,INC) + TERM
      KON = IBIG(SUM,KON)
      IF (ABSF(TERM/SUM) - TOL) 202,202,201
```

```
 201 IF (ABSF(TERM)/TWOK(KON) - 1.E-07) 202,202,2041
2041 KON = KON - LLOW(KON)
     GO TO 204
 202 KON = KON + 1
     IF (MSTAP - LLOW(KON)) 203,203,204
 203 KON = -XMAXOF(ILK,IBIG(TWOK(KON-3072)/SUM,0)) - 1
     GO TO 206
 204 IF (KAPP - IP) 205,205,206
 205 KON = XMAXOF(ILK,IBIG(TERM/SUM,0),IBIG(TWOK(KON-3072)/SUM,0))
 206 D2(IROE,ITH,INC) = SUM
     KCON2(IROE,ITH,INC) = KON
 209 INC = INC + 30
 210 CONTINUE
     DO  217   IRCK=1,NRO1
     DO  217   ITHCK=1,NTH1
     DO  217   ITCK=1,NT1
     INC = ITCK
     DO 217  INK = 1,5
     IF (KCON2(IRCK,ITHCK,INC))216,219,219
 216 INC = INC + 30
 217 CONTINUE
     GO TO 220
 219 CONTINUE
 220 CONTINUE
 221 WRITE OUTPUT TAPE 6,222,COUNT,TALLY,TAP
 222 FORMAT (11H1TERM COUNT,F20.2///39H NO. OF N-SUMS WITH TOO FEW SMAL
    1L TERMS,F15.2///34H NO. OF N-SUMS WITH NO SMALL TERMS,F15.2)
1220 INC = LEADX - 1
     LEAP = 30*INC
 240 DO 241  I = 1,NTH
 241 THETA(I) = 57.2957796*THETA(I)
 245 IF (KODEN - KEY) 247,251,251
 247 IND = 0
     JUMP = -1
     DO 249  INK = 1,5
     DO 248  IX = 1,NT
     INTER = IND + IX
 248 WRITE TAPE 3,((D2(IRO,ITH,INTER),KCON2(IRO,ITH,INTER),IRO = 1,NRO),
    1ITH = 1,NTH)
 249 IND = IND + 30
     REWIND 3
 251 IF (KODEN + KEY - 2) 254,252,3001
 252 DO 253  I = 1,NTH
 253 THETA(I) = PHI + THETA(I)
     JUMP = 1
 254 CONTINUE
C     FINAL OUTPUT
6001 CONTINUE
6010 N1 = NRO
     N2 = NT
     N3 = NTH
     LOC1 = 0
     LOC2 = 12 + LEADX
     LOC3 = 3
     INC1 = 1
     INC2 = 30
     INC3 = 3
     LABEL1 = 0
     LABEL2 = KEY - 1 + KODEN
6011 INC1 = INC1 - N2*INC2
     LOM = 1
```

```
      IF (KODEN - 1) 6512,6514,6516
 6512 WRITE OUTPUT TAPE 6,6513
 6513 FORMAT (18H1DILATATIONAL WAVE)
      GO TO 7019
 6514 WRITE OUTPUT TAPE 6,6515
 6515 FORMAT (11H1SHEAR WAVE)
      GO TO 7019
 6516 WRITE OUTPUT TAPE 6,6017
 6017 FORMAT (46H1SUPERPOSITION OF DILATATIONAL AND SHEAR WAVES)
 7019 WRITE OUTPUT TAPE 6,6018
 6018 FORMAT (2H   ///////////26H THIS OUTPUT HAS THE FORM  //)
 6023 WRITE OUTPUT TAPE 6,6024
 6024 FORMAT (16H RHO,XBAR,THETA ///            //////)
 6025 WRITE OUTPUT TAPE 6,6125
 6125 FORMAT (43H THE SOLUTION MATRIX HAS  SRR,STT,SRT,UR,UT//////)
      DO 6119  I1 = 1,N1
      WRITE OUTPUT TAPE 6,6926
 6926 FORMAT (1H1)
      J1 = I1 + LOC1
      WRITE OUTPUT TAPE 6,6126,TITLE(J1)
 6126 FORMAT (E15.7)
      IF (LABEL1) 6013,6013,6012
 6012 YBAR = (TITLE(J1) - XSTAR) / SVMDUM
      WRITE OUTPUT TAPE 6,6126,YBAR
 6013 DO 6118  I2 = 1,N2
      J2 = I2 + LOC2
      WRITE OUTPUT TAPE 6,6131,TITLE(J2)
 6131 FORMAT (//// E15.7)
      IF (LABEL2) 6015,6015,6014
 6014 YBAR = (TITLE(J2) - XSTAR) / SVMDUM
      WRITE OUTPUT TAPE 6,6126,YBAR
 6015 DO 6117  KA = 1, N3,5
      KO = XMINOF(N3,KA+4)
      LIMA = KA + LOC3
      LIMO = KO + LOC3
      WRITE OUTPUT TAPE 6,6041,(TITLE(J3),J3 = LIMA,LIMO)
 6041 FORMAT (///// 6X,5E21.7 ///)
      LIMA = LOM + INC3*(KA-1)
      LIMO = LOM + INC3*(KO-1)
      DO 6016  INK = 1,5
      IMP = KA
      DO 6115 LIMP = LIMA,LIMO,INC3
      D1(IMP) = D2(LIMP)
      KON = KCON2(LIMP)
      ERROR = ABSF(TWOK(KON))
      IF (ABSF(ERROR - .5) - .5) 226,227,227
  226 KON = -XSIGNF(XINTF(-LOG10F(ERROR)),KON)
      GO TO 228
  227 KON = -0
      IF (ERROR) 228,1227,228
 1227 KON = NACRCY
  228 KCON1(IMP) = KON
 6115 IMP = IMP + 1
      WRITE OUTPUT TAPE 6,6043,(D1(J),KCON1(J),J = KA,KO)
 6043 FORMAT (6X,E17.7,I4,E17.7,I4,E17.7,I4,E17.7,I4,E17.7,I4)
      LIMA = LIMA + 900
 6016 LIMO = LIMO + 900
 6117 CONTINUE
 6118 LOM = LOM + INC2
 6119 LOM = LOM + INC1
C     SELECTING MAXIMAL QUANTITIES
```

```
280 KWO = KWA
290 IF (MAXOUT) 300,350,300
300 WRITE OUTPUT TAPE 6,301
301 FORMAT(17H1MAXIMA OVER XBAR//29H XBAR VALUES ARE BELOW MAXIMA/////
   1///)
    DO   310      ITH=1,NTH1
    THET = THETA(ITH)
    WRITE OUTPUT TAPE 6,302,THET
302 FORMAT (////7H THETA=E14.7 //1H ,16X,3HROE,17X,3HSRR,17X,3HSTT,
   117X,3HSRT)
    DO   310      IRO=1,NRO1
    ROE = RHO(IRO)
    EMAX1=A2(IRO,ITH,1)
    EMAX2=B2(IRO,ITH,1)
    EMAX3=C2(IRO,ITH,1)
    IB1 = LEADX
    IB2 = LEADX
    IB3 = LEADX
    DO   308      ITT=1,NT
    ITT1 = ITT
    AMA1=ABSF(EMAX1)
    AMA2=ABSF(A2(IRO,ITH,ITT1))
    IF(AMA2-AMA1)       304,304,303
303 EMAX1=A2(IRO,ITH,ITT1)
    IB1 = ITT + LEADX -1
304 AMB1=ABSF(EMAX2)
    AMB2=ABSF(B2(IRO,ITH,ITT1))
    IF(AMB2-AMB1)        306,306,305
305 EMAX2=B2(IRO,ITH,ITT1)
    IB2 = ITT + LEADX -1
306 AMC1=ABSF(EMAX3)
    AMC2=ABSF(C2(IRO,ITH,ITT1))
    IF(AMC2-AMC1)     308,308,307
307 EMAX3=C2(IRO,ITH,ITT1)
    IB3 = ITT + LEADX -1
308 CONTINUE
    ETM1 = XBAR(IB1)
    ETM2 = XBAR(IB2)
    ETM3 = XBAR(IB3)
    WRITE OUTPUT TAPE 6,309,  ROE,EMAX1,EMAX2,EMAX3,ETM1,ETM2,ETM3
309 FORMAT (4E20.7/20X,3E20.7/)
310 CONTINUE
    WRITE OUTPUT TAPE 6,311
311 FORMAT(16H1MAXIMA OVER RHO//28H RHO VALUES ARE BELOW MAXIMA//////
   1/)
    DO  319      ITT=1,NT1
    IT = ITT + LEADX - 1
    WRITE OUTPUT TAPE 6,312,XBAR(IT)
312 FORMAT (////6H XBAR=E14.7//15X,5HTHETA,17X,3HSRR,17X,3HSTT,17X,3HS
   1RT)
    DO  319      ITH=1,NTH1
    THET = THETA(ITH)
    EMAX1=A2(1,ITH,ITT)
    IB1 = 1
    EMAX2=B2(1,ITH,ITT)
    IB2 = 1
    EMAX3=C2(1,ITH,ITT)
    IB3 = 1
    DO  317      IRO=1,NRO
    IRO1 = IRO
    AMA1=ABSF(EMAX1)
```

```
      AMA2=ABSF(A2(IRO1,ITH,ITT))
      IF(AM2-AM1)     313,313,3129
 3129 EMAX1=A2(IRO1,ITH,ITT)
      IB1 = IRO
  313 AMB1=ABSF(EMAX2)
      AMB2=ABSF(B2(IRO1,ITH,ITT))
      IF(AMB2-AMB1)     315,315,314
  314 EMAX2=B2(IRO1,ITH,ITT)
      IB2 = IRO
  315 AMC1=ABSF(EMAX3)
      AMC2=ABSF(C2(IRO1,ITH,ITT))
      IF(AMC2-AMC1)     317,317,316
  316 EMAX3=C2(IRO1,ITH,ITT)
      IB3 = IRO
  317 CONTINUE
      ETM1 = RHO(IB1)
      ETM2 = RHO(IB2)
      ETM3 = RHO(IB3)
      WRITE OUTPUT TAPE 6,318,THET,EMAX1,EMAX2,EMAX3,ETM1,ETM2,ETM3
  318 FORMAT (4E20.7/20X,3E20.7/)
  319 CONTINUE
      WRITE OUTPUT TAPE 6,320
  320 FORMAT (18H1MAXIMA OVER THETA//30H THETA VALUES ARE BELOW MAXIMA//
     1//////)
      DO  329      ITT=1,NT1
      IT = ITT + LEADX - 1
      WRITE OUTPUT TAPE 6,321,XBAR(IT)
  321 FORMAT (////6H XBAR=E14.7//17X,3HRHO,17X,3HSRR,17X,3HSTT,17X,3HSRT
     1)
      DO  329      IRO=1,NRO1
      ROE = RHO(IRO)
      EMAX1=A2(IRO,1,ITT)
      IB1 = 1
      EMAX2=B2(IRO,1,ITT)
      IB2 = 1
      EMAX3=C2(IRO,1,ITT)
      IB3 = 1
      DO  327      ITH=1,NTH
      ITH1 = ITH
      AMA1=ABSF(EMAX1)
      AMA2=ABSF(A2(IRO,ITH1,ITT))
      IF(AMA2-AMA1)     323,323,322
  322 EMAX1=A2(IRO,ITH1,ITT)
      IB1 = ITH
  323 AMB1=ABSF(EMAX2)
      AMB2=ABSF(B2(IRO,ITH1,ITT))
      IF(AMB2-AMB1)     325,325,324
  324 EMAX2=B2(IRO,ITH1,ITT)
      IB2 = ITH
  325 AMC1=ABSF(EMAX3)
      AMC2=ABSF(C2(IRO,ITH1,ITT))
      IF(AMC2-AMC1)     327,327,326
  326 EMAX3=C2(IRO,ITH1,ITT)
      IB3 = ITH
  327 CONTINUE
      ETM1 = THETA(IB1)
      ETM2 = THETA(IB2)
      ETM3 = THETA(IB3)
      WRITE OUTPUT TAPE 6,328,ROE,EMAX1,EMAX2,EMAX3,ETM1,ETM2,ETM3
  328 FORMAT (4E20.7/20X,3E20.7/)
  329 CONTINUE
```

```
      EMAX1=A2(1,1,1)
      IB1 = 1
      IB2 = 1
      IB3 = LEADX
      EMAX2=B2(1,1,1)
      IC1 = 1
      IC2 = 1
      IC3 = LEADX
      EMAX3=C2(1,1,1)
      ID1 = 1
      ID2 = 1
      ID3 = LEADX
      DO  335   IRO1=1,NRO1
      DO  335   ITH1=1,NTH1
      DO  335   ITT1=1,NT1
      AMA1=ABSF(EMAX1)
      AMA2=ABSF(A2(IRO1,ITH1,ITT1))
      IF(AMA2-AMA1)       331,331,330
  330 EMAX1=A2(IRO1,ITH1,ITT1)
      IB1 = IRO1
      IB2 = ITH1
      IB3 = ITT1 + LEADX - 1
  331 AMB1=ABSF(EMAX2)
      AMB2=ABSF(B2(IRO1,ITH1,ITT1))
      IF(AMB2-AMB1)     333,333,332
  332 EMAX2=B2(IRO1,ITH1,ITT1)
      IC1 = IRO1
      IC2 = ITH1
      IC3 = ITT1 + LEADX - 1
  333 AMC1=ABSF(EMAX3)
      AMC2=ABSF(C2(IRO1,ITH1,ITT1))
      IF(AMC2-AMC1)     335,335,334
  334 EMAX3=C2(IRO1,ITH1,ITT1)
      ID1 = IRO1
      ID2 = ITH1
      ID3 = ITT1 + LEADX - 1
  335 CONTINUE
      ROEA = RHO(IB1)
      ROEB = RHO(IC1)
      ROEC = RHO(ID1)
      THA = THETA(IB2)
      THB = THETA(IC2)
      THC = THETA(ID2)
      TTA = XBAR(IB3)
      TTB = XBAR(IC3)
      TTC = XBAR(ID3)
      WRITE OUTPUT TAPE 6,336,EMAX1,ROEA,THA,TTA,EMAX2,ROEB,
     1THB,TTB,EMAX3,ROEC,THC,TTC
  336 FORMAT (31H1MAXIMA OVER ALL RHO,THETA,XBAR////
     15H SRR=E14.7,6X,8HFOR RHO=E14.7,6X,6HTHETA=E14.7,6X,5HXBAR=F14.7//
     25H STT=E14.7,6X,8HFOR RHO=E14.7,6X,6HTHETA=E14.7,6X,5HXBAR=E14.7//
     35H SPT=E14.7,6X,8HFOR RHO=E14.7,6X,6HTHETA=E14.7,6X,5HXBAR=F14.7/)
  350 IF (JUMP) 508,1001,351
  351 JUMP = 0
      INC = NX - NT
      DO 354  ICX = 1,NT
      IX = NT + 1 - ICX
      DO 354  IRO = 1,NRO
      IND = 30*IX - 30 + IRO
      DO 353  ITH = 1,NTH
      DO 352  INK = 1,5
```

```
      INTER = IND + LEAP
      D2(INTER) = D2(IND)*AMPFAC
      KCON2(INTER) = KCON2(IND)
  352 IND = IND + 900
  353 IND = IND - 4497
  354 CONTINUE
      IF (INC) 3001,351,357
  357 DO 360  IX = 1,INC
      DO 360  IRO = 1,NRO
      IND = 30*IX - 30 + IRO
      DO 359  ITH = 1,NTH
      DO 358  INK = 1,5
      D2(IND) = 0.
      KCON2(IND) = -1
  358 IND = IND + 900
  359 IND = IND - 4497
  360 CONTINUE
  361 FIX = -1.
      DO 370  INK = 1,5
      IF (INK - 2) 362,363,362
  362 FIX = -FIX
  363 DO 370  IX = 1,NX
      READ TAPE 3,((F(IRO,ITH),JF(IRO,ITH) ,IRO=1,NRO),ITH=1,NTHOLD)
      IF (NFWTH) 3001,565,364
  364 DO 365  ITH = 1,NFWTH
      JTH =KEYTH(ITH)
      INDEX = ITH + NTHOLD
      DO 365  IRO = 1,NRO
      F(IRO,INDEX) = FIX*F(IRO,JTH)
  365 JF(IRO,INDEX) = JF(IRO,JTH)
  565 INDEX = IX + 30*(INK-1)
      DO 368  ITH = 1,NTH
      DO 368  IRO = 1,NRO
      DD = D2(IRO,ITH,INDEX)
      DF = F(IRO,ITH)
      KON = KCON2(IRO,ITH,INDEX)
      JON = JF(IRO,ITH)
      LON = -1
      IF ((KON-XABSF(KON)) + (JON-XABSF(JON))) 367,366,367
  366 LON = 1
  367 KON = XABSF(KON)
      JON = XABSF(JON)
      AB = MAX1F(ABSF(DD*TWOK(KON)),ABSF(DF*TWOK(JON)))
      DD = DD + DF
      KCON2(IRO,ITH,INDEX) = LON * IBIG(AB/DD,0)
  368 D2(IRO,ITH,INDEX) = DD
  370 CONTINUE
      NT = NX
      LEADX = 1
      KODEN = 2
      GO TO 6001
C     ERROR RETURNS
 3001 WRITE OUTPUT TAPE 6,3002,ENG,KAPN,IP,IROE
 3002 FORMAT (15H1 ERROR RETURN  // 5H ENG=E14.7 / 6H KAPN=I6 /
     16H IP=I6 / 6H IROE=I6  )
 3003 CALL PDUMP
      GO TO 1001
 3006 WRITE OUTPUT TAPE 6,3007,LEROR
 3007 FORMAT (19H1ERROR IN INVERSION   //  7H LEROR=I6)
      GO TO 3001
      END
```

```
      SUBROUTINE ZRHSP  (P,E2,E5,E6,E11,Q3TIL,Q3BAR,EPSB,EPST,VMDUM
     1,KTYPE,KAPP,KODSP,CAYIN,T1TOT,TOT,T1T,T2T,REJ,EMJ,EM,ACAPT)
      DIMENSION   REJ(1),EMJ(1),EM(12,1)
      EXPON=EXPF(-Q3TIL)
      TRIG1=SINF(Q3BAR)
      TRIG2=COSF(Q3BAR)
      AR=EPST*(REJ(1)+REJ(3))+(1.0+EPSB)*EMJ(1)-(1.0-EPSB)*EMJ(3)
      AR=0.5*AR
      AI=EPST*(EMJ(1)+EMJ(3))-(1.0+EPSB)*REJ(1)+(1.0-EPSB)*REJ(3)
      AI=0.5*AI
      TILDUM=E6/(E2*P*3.1415927)
      ATR=(-TILDUM)*(Q3BAR*EMJ(2)-Q3TIL*REJ(2))
      ATI=TILDUM*(Q3BAR*REJ(2)+Q3TIL*EMJ(2))
      EM(1,1)=0.0
      EM(2,1)=EXPON*(AR*TRIG2+AI*TRIG1)
      EM(3,1)=(-EXPON)*(ATR*TRIG2+ATI*TRIG1)
      EM(4,1)=0.0
      EM(5,1)=( EXPON)*(AI*TRIG2-AR*TRIG1)
      EM(6,1)=(-EXPON)*(ATI*TRIG2-ATR*TRIG1)
      PI=3.1415927
      PPI=P*PI
      IF(KODSP)    1001,1001,1002
 1001 SP=1.0
      GO TO 1003
 1002 CAPP=KAPP + 10
      PPIP=PPI/CAPP
      SP=SINF(PPIP)/PPIP
 1003 PPI2=PPI/2.0
      GO TO (9001,9002,9003),KTYPE
 9001 AP=(2.0/PPI2)*SINF(PPI2)*SINF(PPI2*E11) * SP
      AP=(2.0/PPI2)*SINF(PPI2)*SINF(PPI2*E11) * SP
      GO TO 9004
 9002 AP=((CAYIN+1.0)/CAYIN)*SINF(PPI*T1T)-SINF(PPI*TOT)
     1-(1.0/CAYIN)*SINF(PPI*T2T)
      AP=(2.0/(PPI**2*T1TOT))*AP
      AP = AP * SP
      GO TO 9004
 9003 COM1=SINF(PPI*T1T)
      COM2=COSF(PPI*T1T)
      EXP1=EXPF(ACAPT*(1.-T1T))
      AP1=(2./(PPI**2*T1TOT))*(COM1-SINF(PPI*TOT))
      AP2=2./PPI
      AP3=ACAPT**2+PPI**2
      AP4=AP2*COM2
      IP=P+.05
      EXP2=((-1.0)**IP)*PPI
      AP5=(2./(AP3*EXP1))*EXP2
      AP6=(2./AP3)*(ACAPT*COM1+PPI*COM2)
      AP=AP1-AP4-AP5+AP6
      AP = AP * SP
 9004 UF=(1.0/PPI)*(E6/(E5*VMDUM*E2))
      DO     9005     I=1,6
 9005 EM(I,1)=EM(I,1)*AP
      EM(3,1)=UF*EM(3,1)
      EM(6,1)=UF*EM(6,1)
      RETURN
      END
```

```
      SUBROUTINE ZRHSM (P,E2,E5,E6,E11,Q4TIL,Q4BAR,EPSB,EPST,VMDUM
     1,KTYPE,KAPP,KODSP,CAYIN,T1TOT,TOT,T1T,T2T,REJ,EMJ,EM,ACAPT)
      DIMENSION REJ(1),EMJ(1),EM(12)
      EXPON = EXPF(-Q4TIL)
      TRIG1 = SINF(Q4BAR)
      TRIG2 = COSF(Q4BAR)
      CR = -EMJ(3)
      CI = REJ(3)
      ADJ = .225079079 * E6 /(P * E2 * SQRTF(VMDUM))
      CTR = ADJ * (Q4TIL*REJ(2) - Q4BAR*EMJ(2))
      CTI = ADJ * (Q4BAR*REJ(2) + Q4TIL*EMJ(2))
      EM(1) = 0.
      EM(2) = EXPON * (CR*TRIG2 + CI*TRIG1)
      EM(3) = -EXPON * (CTR*TRIG2 + CTI*TRIG1)
      EM(4) = 0.
      EM(5) =-EXPON * (CR*TRIG1 - CI*TRIG2)
      EM(6) = EXPON * (CTR*TRIG1 - CTI*TRIG2)
      PPI = 3.14159265*P
      IF(KODSP)   1001,1001,1002
 1001 SP=1.0
      GO TO 1003
 1002 CAPP=KAPP + 10
      PPIP=PPI/CAPP
      SP=SINF(PPIP)/PPIP
 1003 PPI2=PPI/2.0
      GO TO (9001,9002,9003),KTYPE
 9001 AP=(2.0/PPI2)*SINF(PPI2)*SINF(PPI2*E11) * SP
      GO TO 9004
 9002 AP=((CAYIN+1.0)/CAYIN)*SINF(PPI*T1T)-SINF(PPI*TOT)
     1-(1.0/CAYIN)*SINF(PPI*T2T)
      AP=(2.0/(PPI**2*T1TOT))*AP
      AP = AP * SP
      GO TO 9004
 9003 COM1=SINF(PPI*T1T)
      COM2=COSF(PPI*T1T)
      FXP1=FXPF(ACAPT*(1.-T1T))
      AP1=(2./(PPI**2*T1TOT))*(COM1-SINF(PPI*TOT))
      AP2=2./PPI
      AP3=ACAPT**2+PPI**2
      AP4=AP2*COM2
      IP=P+.05
      EXP2=((-1.0)**IP)*PPI
      AP5=(2./(AP3*FXP1))*EXP2
      AP6=(2./AP3)*(ACAPT*COM1+PPI*COM2)
      AP=AP1-AP4-AP5+AP6
      AP = AP * SP
 9004 UF = E6 /(PPI2 * E2 * E5 * SQRTF(2.*VMDUM))
      DO    9005      I=1,6
 9005 EM(I) = EM(I) * AP
      EM(3) = EM(3) * UF
      EM(6) = EM(6) * UF
      RETURN
      END
```

```
      SUBROUTINE RHSP (P,E2,E5,E6,F11,Q3TIL,Q3BAR,EPSB,FPST,VMDUM
     1,KTYPE,KAPP,KODSP,CAXIN,T1TOT,TOT,Y1T,T2T,REJ,EMJ,ACAPT,N,EM)
      DIMENSION    REJ(1),EMJ(1),EM(12,1)
      NEXP=(N+1)/2
      FACTOR=(-1.0)**NEXP
      PIOV2=1.570796327
      PI=3.1415927
      IF(N-2*(N/2))     1,1,2
    1 B1=REJ(N+1)
      B2=EMJ(N+1)
      C1=-EMJ(N+2)
      C2=-EMJ(N)
      C3=REJ(N+2)
      C4=REJ(N)
      D1=REJ(N+3)
      D2=REJ(N-1)
      D3=EMJ(N+3)
      D4=EMJ(N-1)
      GO TO 5
    2 B1=EMJ(N+1)
      B2=-REJ(N+1)
      C1=REJ(N+2)
      C2=REJ(N)
      C3=EMJ(N+2)
      C4=EMJ(N)
      D1= EMJ(N+3)
      D3=-REJ(N+3)
      IF(N-1)    3,3,4
    3 D2=-EMJ(2)
      D4=+REJ(2)
      GO TO  5
    4 D2=+EMJ(N-1)
      D4=-REJ(N-1)
    5 F1=FACTOR*PI
      F2=FACTOR*PIOV2
      ABN=F1*B1
      ATN=F1*B2
      ABCN1=F2*(C1-C2)
      ATCN1=F2*(C3-C4)
      ABSN1=(-F2)*(C1+C2)
      ATSN1=(-F2)*(C3+C4)
      ABCN2=(-F2)*(D1+D2)
      ATCN2=(-F2)*(D3+D4)
      ABSN2=F2*(D1-D2)
      ATSN2=F2*(D3-D4)
      F3=0.31830989
      APNR=EPST*(ABN-ABCN2)+(1.0+EPSB)*ATN+(1.0-EPSB)*ATCN2
      APNR=F3*APNR
      APNI=EPST*(ATN-ATCN2)-(1.0+EPSB)*ABN-(1.0-EPSB)*ABCN2
      APNI=F3*APNI
      CPNR=F3*(EPST*ABSN2-(1.0-EPSB)*ATSN2)
      CPNI=F3*(EPST*ATSN2+(1.0-EPSB)*ABSN2)
      F4=(2.0*E6)/(P*PI*PI*E2)
```

```
      ATPNR=F4*(Q3BAR*ABCN1+Q3TIL*ATCN1)
      ATPNI=F4*(Q3BAR*ATCN1-Q3TIL*ABCN1)
      CTPNR=(-F4)*(Q3BAR*ABSN1+Q3TIL*ATSN1)
      CTPNI=(-F4)*(Q3BAR*ATSN1-Q3TIL*ABSN1)
      EXPON=EXPF(-Q3TIL)
      TRIG1=SINF(Q3BAR)
      TRIG2=COSF(Q3BAR)
      EM(1,1)=0.0
      EM(2,1)=0.0
      EM(3,1)=EXPON*(APNR*TRIG2+APNI*TRIG1)
      EM(4,1)=EXPON*(CPNR*TRIG2+CPNI*TRIG1)
      EM(5,1)=(-EXPON)*(ATPNR*TRIG2+ATPNI*TRIG1)
      EM(6,1)=(-EXPON)*(CTPNR*TRIG2+CTPNI*TRIG1)
      EM(7,1)=0.0
      EM(8,1)=0.0
      EM(9,1)=( EXPON)*(APNI*TRIG2-APNR*TRIG1)
      EM(10,1)=( EXPON)*(CPNI*TRIG2-CPNR*TRIG1)
      EM(11,1)=(-EXPON)*(ATPNI*TRIG2-ATPNR*TRIG1)
      EM(12,1)=(-EXPON)*(CTPNI*TRIG2-CTPNR*TRIG1)
      PI=3.1415927
      PPI=P*PI
      IF(KODSP)   1001,1001,1002
1001  SP=1.0
      GO TO 1003
1002  CAPP=KAPP + 10
      PPIP=PPI/CAPP
      SP=SINF(PPIP)/PPIP
1003  PPI2=PPI/2.0
      GO TO (9001,9002,9003),KTYPE
9001  AP=(2.0/PPI2)*SINF(PPI2)*SINF(PPI2*E11) * SP
      GO TO 9004
9002  AP=((CAYIN+1.0)/CAYIN)*SINF(PPI*T1T)-SINF(PPI*TOT)
     1-((1.0/CAYIN)*SINF(PPI*T2T)
      AP=(2.0/(PPI**2*T1TOT))*AP
      AP = AP * SP
      GO TO 9004
9003  COM1=SINF(PPI*T1T)
      COM2=COSF(PPI*T1T)
      EXP1=EXPF(ACAPT*(1.-T1T))
      AP1=(2./(PPI**2*T1TOT))*(COM1-SINF(PPI*TOT))
      AP2=2./PPI
      AP3=ACAPT**2+PPI**2
      AP4=AP2*COM2
      IP=P+.05
      EXP2=((-1.0)**IP)*PPI
      AP5=(2./(AP3*EXP1))*EXP2
      AP6=(2./AP3)*(ACAPT*COM1+PPI*COM2)
      AP=AP1-AP4-AP5+AP6
      AP=AP1-AP4-AP5+AP6-AP7
      AP = AP * SP
9004  UF=(1.0/PPI)*(E6/(E5*VMDUM*E2))
      DO    9005     I=1,12
9005  EM(I,1)=EM(I,1)*AP
      EM(5,1)=UF*EM(5,1)
      EM(6,1)=UF*EM(6,1)
      EM(11,1)=UF*EM(11,1)
      EM(12,1)=UF*EM(12,1)
      RETURN
      END
```

```
      SUBROUTINE RHSM (P,E2,E5,E6,E11,Q4TIL,Q4BAR,EPSA,EPST,VMDUM
     1,KTYPE,KAPP,KODSP,CAYIN,T1TOT,TOT,T1T,T2T,REJ,EMJ,ACAPT,N,EM)
      DIMENSION   REJ(1),EMJ(1),EM(12,1)
      NEXP=(N+1)/2
      FACTOR=(-1.0)**NEXP
      PIOV2=1.570796327
      PI=3.1415927
      IF(N-2*(N/2))    1,1,2
    1 C1=-EMJ(N+2)
      C2=-EMJ(N)
      C3=REJ(N+2)
      C4=REJ(N)
      D1=REJ(N+3)
      D2=REJ(N-1)
      D3=EMJ(N+3)
      D4=EMJ(N-1)
      GO TO 5
    2 C1=REJ(N+2)
      C2=REJ(N)
      C3=EMJ(N+2)
      C4=EMJ(N)
      D1= EMJ(N+3)
      D3=-REJ(N+3)
      IF(N-1)   3,3,4
    3 D2=-EMJ(2)
      D4=+REJ(2)
      GO TO  5
    4 D2=+EMJ(N-1)
      D4=-REJ(N-1)
    5 F1=FACTOR*PI
      F2=FACTOR*PIOV2
      ABCN1=F2*(C1-C2)
      ATCN1=F2*(C3-C4)
      ABSN1=(-F2)*(C1+C2)
      ATSN1=(-F2)*(C3+C4)
      ABCN2=(-F2)*(D1+D2)
      ATCN2=(-F2)*(D3+D4)
      ABSN2=F2*(D1-D2)
      ATSN2=F2*(D3-D4)
      F3 = .6366198
      APNR = F3 *ATSN2
      APNI = -F3 *ABSN2
      CPNR = F3 *ATCN2
      CPNI = -F3 * ABCN2
      F4 = .143289793 * E6 /(P * E2 * SQRTF(VMDUM))
      ATPNR = F4 * (Q4BAR*ABSN1 + Q4TIL*ATSN1)
      ATPNI = F4 * (Q4BAR*ATSN1 - Q4TIL*ABSN1)
      CTPNR = F4 * (Q4BAR*ABCN1 + Q4TIL*ATCN1)
      CTPNI = F4 * (Q4BAR*ATCN1 - Q4TIL*ABCN1)
      EXPON = EXPF(-Q4TIL)
      TRIG1 = SINF(Q4BAR)
      TRIG2 = COSF(Q4BAR)
      EM(1,1)=0.0
      EM(2,1)=0.0
      EM(3,1)=EXPON*(APNR*TRIG2+APNI*TRIG1)
      EM(4,1)=EXPON*(CPNR*TRIG2+CPNI*TRIG1)
      EM(5,1)=(-EXPON)*(ATPNR*TRIG2+ATPNI*TRIG1)
      EM(6,1)=(-EXPON)*(CTPNR*TRIG2+CTPNI*TRIG1)
```

```
      EM(7,1)=0.0
      EM(8,1)=0.0
      EM(9,1)=( EXPON)*(APNI*TRIG2-APNR*TRIG1)
      EM(10,1)=( EXPON)*(CPNI*TRIG2-CPNR*TRIG1)
      EM(11,1)=(-EXPON)*(ATPNI*TRIG2-ATPNR*TRIG1)
      EM(12,1)=(-EXPON)*(CTPNI*TRIG2-CTPNR*TRIG1)
      PPI=P*PI
      IF(KODSP)   1001,1001,1002
 1001 SP=1.0
      GO TO 1003
 1002 CAPP=KAPP + 10
      PPIP=PPI/CAPP
      SP=SINF(PPIP)/PPIP
 1003 PPI2=PPI/2.0
      GO TO (9001,9002,9003),KTYPE
 9001 AP=(2.0/PPI2)*SINF(PPI2)*SINF(PPI2*E11) * SP
      GO TO 9004
 9002 AP=((CAYIN+1.0)/CAYIN)*SINF(PPI*T1T)-SINF(PPI*TOT)
     1-(1.0/CAYIN)*SINF(PPI*T2T)
      AP=(2.0/(PPI**2*T1TOT))*AP
      AP = AP * SP
      GO TO 9004
 9003 COM1=SINF(PPI*T1T)
      COM2=COSF(PPI*T1T)
      EXP1=EXPF(ACAPT*(1.-T1T))
      AP1=(2./(PPI**2*T1TOT))*(COM1-SINF(PPI*TOT))
      AP2=2./PPI
      AP3=ACAPT**2+PPI**2
      AP4=AP2*COM2
      IP=P+.05
      EXP2=((-1.0)**IP)*PPI
      AP5=(2./(AP3*EXP1))*EXP2
      AP6=(2./AP3)*(ACAPT*COM1+PPI*COM2)
      AP=AP1-AP4-AP5+AP6
      AP = AP * SP
 9004 UF = E6 /(PPI2 * F2 * E5 * SQRTF(2.*VMDUM))
      DO    9005     I=1,12
 9005 EM(I,1)=EM(I,1)*AP
      EM(5,1)=UF*EM(5,1)
      EM(6,1)=UF*EM(6,1)
      EM(11,1)=UF*EM(11,1)
      EM(12,1)=UF*EM(12,1)
      RETURN
      END
```

```
      SUBROUTINE LINSYS (A,Y,M,L)
      DIMENSION A(12,12),Y(12)
      M1 = M - 1
      DO 150  K = 1,M1
      KP = K + 1
      X = 0.
      DO 110  I = K,M
      IF (X - ABSF(A(I,K))) 100,110,110
  100 X = ABSF(A(I,K))
      L = I
  110 CONTINUE
      IF (X) 120,120,130
  120 L = 0
      GO TO 190
  130 DO 140  J = 1,M
      X = A(K,J)
      A(K,J) = A(L,J)
  140 A(L,J) = X
      X = Y(K)
      Y(K) = Y(L)
      Y(L) = X
      DO 150  I = KP,M
      X = A(I,K) / A(K,K)
      Y(I) = Y(I) - Y(K)*X
      DO 150  J = KP,M
  150 A(I,J) = A(I,J) - A(K,J)*X
      IF (A(M,M)) 160,120,160
  160 Y(M) = Y(M)/A(M,M)
      K = M
      DO 180  I = 1,M1
      X = 0.
      KP = K
      K = K - 1
      DO 170  J = KP,M
  170 X = X + A(K,J)*Y(J)
  180 Y(K) = (Y(K) - X) / A(K,K)
  190 RETURN
      END
```

```
      SUBROUTINE TILDE (K,Q,ROE,THETA,REJTIL,EMJTIL,
     1REYTIL,EMYTIL,ENG)
      DIMENSION  REJTIL(1),EMJTIL(1),REYTIL(1),EMYTIL(1)
      IMORE=1
      IF(THETA)  2,1,2
    1 IMORE=2
    2 N = ROE
      IF ACCUMULATOR OVERFLOW 2001,2001
 2001 N = 10 + XMAXOF(K,N+20)
  810 REJTIL(N+2)=0.0
      EMJTIL(N+2)=0.0
      REJTIL(N+1) = 1.E-18
      EMJTIL(N+1) = 0.
 2003 RQ=ROE*Q
      F1=1.0
      GO TO (3,4),IMORE
    3 X=RQ*COSF(THETA)
      Y=RQ*SINF(THETA)
      F1=X/RQ
      F2=Y/(RQ*ROE)
    4 F1=F1/ROE
      QSQ4=Q**2/4.0
      DO 16   I=1,N
      L=N+1-I
      EL=L
      REJTIL(L)=F1*REJTIL(L+1)-(QSQ4*REJTIL(L+2))/(EL*(EL+1.0))
      EMJTIL(L) = 0.
      GO TO (5,6),IMORE
    5 REJTIL(L)=REJTIL(L)+F2*EMJTIL(L+1)
      EMJTIL(L)=F1*EMJTIL(L+1)-F2*REJTIL(L+1)-(QSQ4*EMJTIL(L+2))
     1/(EL*(EL+1.0))
    6 IF (ABSF(REJTIL(L)) + ABSF(EMJTIL(L)) - 1.E30) 16,16,2004
 2004 DO 2005  J = L,N
      REJTIL(J) = REJTIL(J) * 1.E-30
 2005 EMJTIL(J) = EMJTIL(J) * 1.E-30
   16 CONTINUE
      ENG=0.0
      CALL JTILO (Q,ROE,THETA,FP,GP,ENG)
      IF(ENG)  7,7,28
    7 GO TO  (8,9),IMORE
    8 A = REJTIL(1)
      B = EMJTIL(1)
      HIGH = MAX1F(ABSF(A),ABSF(B))
      AH = A/HIGH
      BH = B/HIGH
      DENOM = A*AH + B*BH
      A = (FP*AH + GP*BH) / DENOM
      B = (GP*AH - FP*BH) / DENOM
      GO TO 10
    9 A=FP/REJTIL(1)
   10 DO  12   I=1,N
      DUM1=REJTIL(I)
      REJTIL(I)=A*DUM1
      GO TO (11,12),IMORE
   11 DUM2=EMJTIL(I)
      REJTIL(I)=REJTIL(I)-B*DUM2
      EMJTIL(I)=A*DUM2+B*DUM1
   12 CONTINUE
      CAPA=-0.1159315157+LOGF(RQ)
      SUMRE=(0.6366197772)*(CAPA*FP-THETA*GP)
      SUMIM=(0.6366197772)*(CAPA*GP+THETA*FP)
      GAM=1.273239544
      FOUMMY=QSQ4
```

```
      FACTOR=FDUMMY/2.0
      NOV2=(N+1)/2
      KTEST=1
      LTEST=1
      DO 24    M=1,NOV2
      M2=2*M+1
      TERM=GAM*FACTOR
      GO TO (15,1701),KTEST
   15 TERMRE=TERM*REJTIL(M2)
      IF(ABSF(TERMRE)-1.0E-11)  46,46,17
   46 KTEST=2
      GO TO 1701
   17 SUMRE=SUMRE+TERMRE
 1701 GO TO (1702,22),LTEST
 1702 GO TO (19,18),IMORE
   18 LTEST=2
      GO TO 22
   19 TERMIM=TERM*EMJTIL(M2)
      IF(ABSF(TERMIM)-1.0E-11)  20,20,21
   20 LTEST=2
      GO TO 22
   21 SUMIM=SUMIM+TERMIM
   22 IF(KTEST+LTEST-4)  23,25,25
   23 BM=(M2+1)*M2
      FACTOR=(FACTOR*FDUMMY)/BM
      EM=M+1
      GAM=(-GAM)*((EM-1.0)/EM)
   24 CONTINUE
      ENG=3.0
      GO TO 28
   25 REYTIL(1)=SUMRE
      EMYTIL(1)=SUMIM
      GO TO (27,26),IMORE
   26 TOP=QSQ4*SUMRE*REJTIL(2)-Q/(3.141592653*RQ)
      REYTIL(2)=TOP/REJTIL(1)
      EMYTIL(2) = 0.
      GO TO 2009
   27 BOTTOM=REJTIL(1)**2+EMJTIL(1)**2
      T1=REJTIL(1)*REJTIL(2)+EMJTIL(1)*EMJTIL(2)
      T2=REJTIL(1)*EMJTIL(2)-EMJTIL(1)*REJTIL(2)
      T3=1.0/(3.14159265*ROE*RQ)
      T4=X*REJTIL(1)-Y*EMJTIL(1)
      T5=Y*REJTIL(1)+X*EMJTIL(1)
      REYTIL(2)=(QSQ4*REYTIL(1)*T1-QSQ4*EMYTIL(1)*T2
     1-T3*T4)/BOTTOM
      EMYTIL(2)=(QSQ4*REYTIL(1)*T2+QSQ4*EMYTIL(1)*T1
     1+T3*T5)/BOTTOM
 2009 IF ACCUMULATOR OVERFLOW 2010,28
 2010 ENG = -2.
   28 RETURN
      END
```

```
      SUBROUTINE BESSEL (K,Q,THETA,REJ,EMJ,ENG)
      DIMENSION  REJ(1),EMJ(1)
   10 FORMAT (24H OVERFLOW IN BESSEL. Q =E15.7,9H  THETA =E15.7,7H KAPN
     1=I3)
      IF ACCUMULATOR OVERFLOW 100,100
  100 KTEST1 = Q + 20.
      KTEST2=K
      IF(KTEST1-KTEST2)    1,1,2
    1 N=KTEST2+10
      GO TO 3
    2 N=KTEST1+10
    3 REJ(N+2)=0.0
      EMJ(N+2)=0.0
      REJ(N+1)=1.0E-37
      EMJ(N+1)=0.0
      X=Q*COSF(THETA)
      Y=Q*SINF(THETA)
      N1=N+1
      QSQ=Q*Q
      Q1=(2.0*X)/QSQ
      Q2=(2.0*Y)/QSQ
      DO   4        I=1,N
      L=N1-I
      EL=L
      REJ(L)=EL*(Q1*REJ(L+1)+Q2*EMJ(L+1))-REJ(L+2)
      EMJ(L)=EL*(Q1*EMJ(L+1)-Q2*REJ(L+1))-EMJ(L+2)
      IF (ABSF(REJ(L)) + ABSF(EMJ(L)) - 1.0E30) 4,13,13
   13 DO 14  J = L,N
      REJ(J) = REJ(J) * 1.0E-30
   14 EMJ(J) = EMJ(J) * 1.0E-30
    4 CONTINUE
      ENG=0.0
C     CALCULATE ALPHA
      KF=1
      KG=1
      RTEST=0.00000005
      FP=REJ(3)
      GP=EMJ(3)
      DO 9007  ISUM = 5,N,2
      GO TO (9001,9003),KF
 9001 FP=FP+REJ(ISUM)
      RELF=ABSF(REJ(ISUM)/FP)
      IF(RELF-RTEST)  9002,9002,9003
 9002 KF=2
 9003 GO TO  (9004,9006),KG
 9004 GP=GP+EMJ(ISUM)
      RELG=ABSF(EMJ(ISUM)/GP)
      IF(RELG-RTEST)  9005,9005,9006
 9005 KG=2
 9006 IF(KG+KF-4)   9007,9008,9008
 9007 CONTINUE
      ENG = -2.
      GO TO 200
 9008 A =REJ(1)+2.0*FP
      B =EMJ(1)+2.0*GP
 5005 HIGH = MAX1F(ABSF(A),ABSF(B))
      X = A/HIGH
      Y = B/HIGH
      DENOM = A*X + B*Y
      A = X/DENOM
      B = Y/DENOM
 6006 DO    6      I=1,N
      DUM1=REJ(I)
      DUM2=EMJ(I)
      REJ(I)=A*DUM1+B*DUM2
      EMJ(I)=A*DUM2-B*DUM1
    6 CONTINUE
      IF ACCUMULATOR OVERFLOW 200,7
  200 WRITE OUTPUT TAPE 6,10,Q,THETA,K
      ENG = -1.
    7 RETURN
      END
```

```
      SUBROUTINE JTILO (QP,ROEP,THETAP,FP,GP,ERROR)
      DIMENSION  Q(1),ROE(1),THETA(1),F(1),G(1)
      Q(1)=QP
      ROE(1)=ROEP
      THETA(1)=THETAP
      Q(2)=0.0
      ROE(2)=0.0
      THETA(2)=0.0
      KTEST=1
      LTEST=1
      IF(THETA)    2,1,2
    1 LTEST=2
    2 F=1.0
      G=0.0
      A=-1.0
      B=Q*ROE/2.0
      DO    5     I=1,1000
      P=1.0
      EI=I
      DO    3     J=1,I
      EJ=J
      P=P*(B/EJ)*(B/EJ)
    3 CONTINUE
      GO TO (301,304),KTEST
  301 PF=P*COSF(2.0*EI*THETA)
      IF(ABSF(PF)-0.1**11)   302,302,303
  302 KTEST=2
      GO TO  304
  303 F=F+A*PF
  304 GO TO (305,308),LTEST
  305 PG=P*SINF(2.0*EI*THETA)
      IF(ABSF(PG)-0.1**11)   306,306,307
  306 LTEST=2
      GO TO 308
  307 G=G+A*PG
  308 A=-A
      IF(KTEST+LTEST-4)    5,503,503
    5 CONTINUE
      GO TO (501,502),KTEST
  501 ERROR=1.0
      GO TO  504
  502 ERROR=2.0
      GO TO  504
  503 FP=F(1)
      GP=G(1)
  504 RETURN
      END
```

```
      FUNCTION IBIG(A,N)
      IF (A) 100,140,100
100   Y = 1.44269504 * LOGF(ABSF(A))
      IF (Y) 120,110,110
110   Y = Y + 1.
120   M = Y
      M = 128 * M  + 16384
      IF (M - N) 140,140,130
130   L = N - 128*(N/128)
      IBIG = L + M
      GO TO 150
140   IBIG = N
150   RETURN
      END
```

```
      FUNCTION LLOW(K)
      LLOW = K - 128*(K/128)
      RETURN
      END
```

```
      FUNCTION TWOK(K)
      L = K / 128  - 128
      TWOK = 2.**L
      RETURN
      END
```

Unclassified
Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Grumman Aircraft Engineering Corporation Bethpage, N. Y.	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP

3. REPORT TITLE
DYNAMIC STRESSES IN A THICK ELASTIC CYLINDER SUBJECT TO TRANSIENT PRESSURE LOADINGS
Volume II: Discussion of Computer Program

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)
1 June 1963 through 16 March 1965

5. AUTHOR(S) (Last name, first name, initial)
Nolan, Frank

6. REPORT DATE September 1965	7a. TOTAL NO. OF PAGES 76	7b. NO. OF REFS
8a. CONTRACT OR GRANT NO. AF 29(601)-5993 b. PROJECT NO. 5710 c. Subtask 13.148 d.	9a. ORIGINATOR'S REPORT NUMBER(S) AFWL-TR-65-20, Vol. II. 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)	

10. AVAILABILITY/LIMITATION NOTICES
Qualified users may obtain copies of this report from DDC.
Distribution is limited because of the technology discussed in this report.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY Air Force Weapons Laboratory (WLDC) Kirtland Air Force Base, New Mexico

13. ABSTRACT

The response of a hollow circular cylindrical shell of arbitrary thickness, in either an elastic or a viscoelastic medium, to transient dilatational and shear waves (and their superposition) is presented. The solution is valid within the scope of the linear theory of elasticity or viscoelasticity. The technique for obtaining the solution relies upon 1) the construction of a train of incident pulses from steady-state components, where each pulse represents the time history of the transient stress in the incident wave, and 2) the existence of a physical mechanism that, between pulses, restores the disturbed particles of the cylinder and the surrounding medium to an unstrained state of rest.

The influence on the cylinder response of the following factors is discussed: liner thickness, cylinder-medium impedance mismatch, viscoelasticity in the medium, and incident wave form (step pulse, rectangular, triangular, linear rise-exponential decay).

DD FORM 1 JAN 64 1473

Unclassified
Security Classification

Unclassified

Security Classification

14. KEY WORDS	LINK A ROLE	LINK A WT	LINK B ROLE	LINK B WT	LINK C ROLE	LINK C WT
Hollow circular cylindrical shell of arbitrary thickness						
Elastic and viscoelastic medium						
Transient dilatational and shear waves						
Liner displacements and stresses						
Underground structures - effects of blast						

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization (*corporate author*) issuing the report.

2a. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2b. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year; or month, year. If more than one date appears on the report, use date of publication.

7a. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7b. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8a. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8b, 8c, & 8d. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9a. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9b. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers (*either by the originator or by the sponsor*), also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through

________________________________."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through

________________________________."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through

________________________________."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring (*paying for*) the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS), (S), (C), or (U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, rules, and weights is optional.

AFSC (MAFB ...) [illegible]

Unclassified

Security Classification